# ATOM NIEUWS

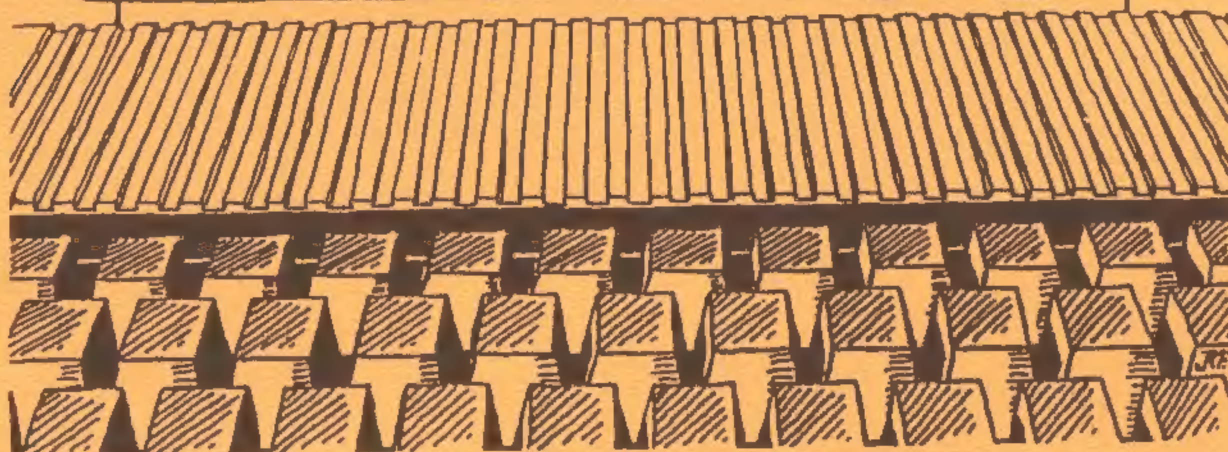
JAARGANG : 13
NUMMER : 1
DISKNR. : DOS 1+2




WAT IS

HET MENU

VAN MORGEN



FEDERATIE VAN ATOMCLUBS NEDERLAND - BELGIE.

Voorzitter :

P.v.Kuik
Zuideinde 54-a
1843 JP Groot-Schermer
tel. 02997-1902

Secretaris:

J.Hartog
Keyenbergseweg 60
6871 WK Renkum
tel. 08373-13757

Penningmeester:

T.Rutten
Berkenlaan 24
3737 RN Groenekan
tel.03461-3495

Contributie 1994 : fl. 25,00 : Atom Computerclub : Giro 5244293.

Redactie Atom Nieuws

B.Tossaint 043-431675
W.Truijen 09-3289564792
R.Leurs 046-370650

Redactieadres A.N.

B.Tossaint
Fatimaplein 85
6214 TW Maastricht
tel. 043-431675

Ledenadministratie

T.Rutten
Berkenlaan 24
3737 RN Groenekan
tel. 03461-3495

UITERSTE DATUM INLEVERING KOPY VOOR NR. 13-2 : 1 JUNI 1994

Clubwinkel

J.Hartog
Keyenbergseweg 60
6871 WK Renkum
tel. 08373-13757

ATOM-TEL

E.Gijssel
Ruysdaelstraat 6
4462 AD Goes
tel. 01100-32419

De Clubwinkel :

| | |
|---|---|
| 80-koloms-video-kaart excl. onderdelen | fl. 5,00 |
| Combikaart 91 versie 1 : zie SPS-Printservice | |
| Z-80-kaart voor CP/M , exlusief onderdelen | fl. 10,00 |
| ACORN NIEUWS 1982, 97 pagina's samenvatting | fl. 1.00 |
| ATOM NIEUWS jaargang 1983 , +/- 450 pag. | fl. 1.00 |
| ATOM NIEUWS Jaargang 1984 , +/- 650 pag. | fl. 1,00 |
| ATOM NIEUWS Jaargang 1985 , +/- 650 pag. | fl. 1,00 |
| ATOM NIEUWS Jaargang 1986 , +/- 500 pag. | fl. 1,00 |
| ATOM NIEUWS Jaargang 1987 , +/- 300 pag. | fl. 1,00 |
| ATOM-WARE : deel 1 : Atom-basic interpreter , 98 pag. | fl. 1,00 |
| ATOM-WARE : deel 2 : Atom-disk operating syst.68 pag. | fl. 1,00 |
| ATOM-WARE : deel 3 : Monitor operating system 80 pag. | fl. 1,00 |

Levering geschiedt via uw regionale penningmeester, of rechtstreeks, via de penningmeester van de federatie . Bij rechtstreekse bestelling stort U het bedrag van het gewenste artikel , vermeerderd met fl. 4,00 portokosten , op de giro van de federatie , met de vermelding van de naam van het artikel en uw lidmaatschapsnummer.

# I N H O U D S O P G A V E



DE EERSTE 25 ATOM-IN-PC KAARTEN ZIJN AL VERKOCHT

EN DOOR 11 PERSONEN IS AL INGESCHREVEN OP DE VOLG-SERIE!!!

ZULLEN WE DE 50 HALEN ???

Als u oud-leden weet , die interesse zouden hebben voor ATOM-IN-PC dan lijkt het zinvol deze te attenderen op onze aanbiedingen. ( in voor hen aan te passen vorm , uiteraard ).

TOT NU TOE ZIJN ER 50 LEDEN DIE IN 1994 MEE DOORGAAN !.

Met in ieder geval nog een aantal achterblijvers,
de 60 halen we waarschijnlijk wel !.

*********** 23 APRIL ***** ATOM-LANDDAG 1994 ***********

zie ook pag 43.

In ons volgende nummer o.a. :
De korste verbinding tussen twee lijnen , v. L.Bijnagte
De Atom in Pc, maar dan geheel in PC-software, v. R.Bronsdijk

VAN DE REDACTIE.

Alweer een jaar erbij, jaargang 13 maken we nu.
Een fraai getal, met zo veel verschillende betekenissen.

In ieder geval blijkt het een opvallend jaar te zijn, zie de verschijning van de Atom in Pc voor iedereen ,enerzijds, en anderzijds de Atom als software-emulatie , zoals die inmiddels in ontwikkeling is door Dick Bronsdijk.
In het volgende nummer kunt u zijn informatie verwachten.

Een merkwaardige coïncidentie ; een hardware en een software Atom voor de Pc. bijna tegelijk !.

En dan verschijnt daar Leendert Bijnagte weer aan het Atom-firmament en pakt meteen , zoals te zien in dit nummer , op een forse manier uit.
Window drivers voor de Atom ,etc, zoveel,dat we een artikel van hem al naar het volgende nummer hebben doorgeschoven.

Bij dit nummer worden geleverd: twee regioschijven in Dos-form.

De ene vol met Atom-programmatuur aangepast door Willy Truyen aan de ATOM-IN-PC, de ander met de nieuwe versie van de systeem-soft voor de Atom-in-Pc, inclusief de inhoud van de bijbehorende Eprom-vulling. Met de oude werkt het ook wel, maar met de nieuwe beter.

Er zal wellicht een andere regeling moeten komen voor de verspreiding van de software, nu regio's in de oude vorm bijna niet meer bestaan , onderwerp voor de jaarvergadering
De schijven gaan nu alleen nog naar : Sj.Geene, P.v.Kuyk, J.Lernout, T.Rutten en Th. Waayer.
(Men kan de software uiteraard ook bij W.Truyen aanvragen).

Er is inmiddels een kaart die het kennelijk niet doet, naar de oorzaak zal natuurlijk worden gezocht.
Een ander gaat soms op onderdelen in de fout, ook daarna....
Overigens , het moet wel gezegd: de printout van de programmatuur zoals die in de Pc wordt opgeslagen , beslaat inmiddels een boek van 3 cm dikte in A4-formaat. Vooralsnog weet alleen Roland daarin de weg.

En anderen gaan onverdroten verder, Sj.Geene komt eerstdaags op het tapijt met zijn complete ruisvrije kleurenkaart.

Al met al , een hoopvol begin van het nieuwe jaar.

Namens de redactie
Bruno Tossaint.

Terug van eigenlijk niet weggeweest

Na enige jaren van afwezigheid binnen de AAC, toch weer terug gekeerd tot de echte bron: de Acorn Atom Club. Op de landelijke Big Ben Club dag, was een enthousiaste afvaardiging van de AAC aanwezig en zie daar mijn rustende liefde voor de Atom laaide spontaan op. En daar ik een (voor)geprogrammeerde noodkreet van de redactie zag, dacht ik: kom ik klim in de printer en probeer wat kopij aan te maken.
Ik ben van plan de draad, van die regelmatige stukjes, weer op te pakken. Misschien kent u nog de Assemblercursus in de Atomniews, of de printerdriver voor de Microline 80. Dat waren nog eens tijden. Het waren de tijden dat je gewoon kon begrijpen wat er meestal in de machine gebeurde. Een clear 4 dump op een Microline 80. Goed voor een A4 vol. Tegenwoordig op m'n HP laserjet: nauwelijks meer dan een postzegel.
Vele avonden lang zitten studeren op de ROM listing, gemaakt op een van een werkgever geleende printer. Verbijstering over het slimme wat in de Atom bijeengebracht is, voor een, ook toen, redelijke prijs.
Toch is er veel veranderd sindsdien. Apparatuur wordt sneller, programmatuur complexer en het is welhaast niet meer mogelijk om 'alles' van je computer te weten. Vandaar dat ik best met weemoed terug denk aan die tijd waarin je gewoon even naar een 'Atom-maatje' waar dan ook in Nederland, belde en prompt een antwoord kreeg. Zie dat eens in verhouding met de PC rage. Het lijkt potjandrie wel op postzegels verzamelen, met enig verschil dat de afbeeldingen de programma's zijn geworden en de kartelrantjes de versienummers van die programma's. De vergelijking gaat verder: Als men eenmaal zo'n zegeltje heeft, is het bepaald niet de bedoeling deze te vergezellen van bijbehorende enveloppe en te sturen naar Timboektoe. Nee, met uitgewisselde software moet je vooral niets doen: gebruik het als ruilobject voor weer een nieuwe versie of nog een tekstverwerker die nog meer kan wat je niet nodig heb. Natuurlijk, ik draaf licht door, maar de trend is duidelijk.
Nee, geef mij dan maar de vertrouwde Atom: Open, helder en gewoon leuk. Toch ben ik in mijn Atom-afwezige periode, verder gegaan. Als het u niet vermoeit: hier een kleine bloemlezing. Niet op te zeggen van Goh, kijk eens wat ik kan, maar meer van: Goh, daar zou ik wel eens wat meer van willen weten. Als dat laatste zo is: laat het me weten. Bel gerust, maar houdt er rekening met dat vader en Atom na 22:30 rusten.

De opvolger van de Atom werd een BBC. Inmiddels afgestudeerd, dus qua financiën paste dit eindelijk. Werkelijk een voortreffelijke machine, een echte super Atom. Best nog wel leuke dingen mee gedaan. Onder andere, naast vele uitbredingen in de sideways ROM, een interface gemaakt tussen twee van die machines, waar bij het mogelijk was om via een commando het geheugen uit de andere machine over te kopiëren naar het eigen geheugen. Het geheel werkte op basis van de 6522 en later de 6821, en, uiteraard, onder interrupt, dus de andere machine liep vrolijk door, terwijl het geheugen gecloond werd. Het systeem werd door een 06-xx exploitant gebruikt. Niet zo'n lijn waar je met beslagen bril naar allerlei gesampelde weinig-om-het-lijf verhaaltjes moet luisteren, maar een Windline, een lijn die de actuele weer situatie van een aantal lokaties geeft. Met name voor surfers en zeilers schijnt dit handig te zijn, daar deze anders voor Jan Doedel naar de kust rijden. Het ene station was helemaal volgebouwd met signaleringsapparatuur, de andere gekoppeld aan een spraak systhesyzer. Later is dit overgezet naar Archimedes computers. Dat was ook mijn opvolger voor de BBC. Dat er bij ook nog een IBM-PC verscheen, is niet verbazingwekkend. Fantastische machine die Archimedes. Iets te. Dat wil zeggen, alles kan, maar je hoeft niet te weten waarom. Als je iets wil, dan roep je een functie aan, en dan krijg je een voorgedefinieerd antwoord. Over het geheugen hoef je je, kan je je zelfs niet druk maken. Je mag per gratie vragen stellen over het geheugen en daarmee uit. Niet echt iets voor Atom jongens dus.

En wat doe ik verder zoal: Ik werk als technical-consultant bij Europa's grootste belegger. Ben daar verantwoordelijk voor de technische infrastructuur van het bedrijf.

Daarnaast een eigen bedrijf, die turn-key projecten doet. Is veel systeemontwikkeling. Met name op het gebied van Windows en OS/2.

Op de Big Ben dag heb ik het prachtige kind van Roland gezien: de Atom op een echte IBM kaart. Uiteraard ingeschreven op zo'n ding: Het is me nu duidelijk waar om er PC's zijn: omdat die kaart gemaakt moest worden. Nou is die software die Roland geschreven heeft prima, wat ik echter ga doen, is die kaart onder Windows brengen. Dus naast Excel, Word en CorelDraw: de Atom. Ik verheug me er al op. Dat dat trouwens een niet echt eenvoudige klus is, dat moge duidelijk zijn. En ik ben dan ook sterk van plan u in elke Atom nieuws te vermoeien met de voortgang en de aspecten van de software in combinatie met de Atomkaart. De kaart is van een zodanige kwaliteit dat die aandacht gerechtvaardigd is. Misschien ook nog wel werkend voor de software Atom op de PC gemaakt door Dick Bronsdijk. Ook al zo'n produkt wat op moet voor een lintje.

Wat zit er nog meer in de pen. Een verhaal over hoe maak ik programma's. Programmeren is meer dan het intypen van een listing. En omdat ik heel veel hardware ideeën in de Atom nieuws van afgelopen jaren gezien heb, lijkt het me leuk om het aspect te belichten over hoe ontwerp je programma's die uit soms meer dan 8000 regels bestaan, en hoe hou je het overzicht.

Een ander verhaal over hoe trek je lijnen in assembler. Hoe zorg je dat dat een beetje vlot gaat. Wat zijn de achterliggende algoritmen.

Datacompressie. Veel gehoord. Veel gebruikt. Veel verguisd. Deed de hele wereld het maar op z'n Atom's. Een tekstverwerker in een 4k ROM inclusief nieuwe beeldschermroutines. Alleen de read.me file van de nieuwe tekstverwerker is al 10x groter. Toch een verhaal over hoe je zaken kan comprimeren. Mischien zelfs met voorbeeld.

En misschien neemt iemand de euvele moed om in te telefoon te klauteren en te zeggen, joh schrijf eens iets over...

Nou, tot horens dan maar...

Leendert Bijnagte, 03410 30804



# ATOM TERMINAL

# PROGRAMMA

# VERSIE 2.00




door roland leurs

**Inleiding**

Nu de Atom-in-PC kaart in gebruik genomen is bleek dat het terminal programma toch nog vatbaar was voor enkele uitbreidingen en verbeteringen.

Door het ontbreken van deze uitbreidingen was het onder andere niet mogelijk om in de standaard Atom schermroutines (32*16) een karakter naar de printer te sturen. In ATOM.COM versie 2.xx is een commando toegevoegd dat, als de printer aan staat, een karakter naar de printer stuurt. Dit commando is vergelijkbaar met de routine #FEFB in het Atom operating systeem.

Verder zijn toegevoegd:

- nieuwe controlcode's
- nieuwe commando's voor extended video modi
- ondersteuning beeldscherm pagina's in tekst mode
- communicatie met andere programma's in de PC

Hoe een en ander werkt staat uitgebreid beschreven in dit supplement. Dit supplement is wat meer naar de praktijk gericht dan de eerste handleiding met het doel om een en ander wat duidelijker uit te leggen.

Voor commentaar houd ik me nog steeds aanbevolen!

**Aanvullingen en correcties op de eerste handleiding.**

Allereerst een ontbrekend woord in de inleiding waar velen al over gevallen zijn: op de vijfde regel is aan het eind het woordje 'zich' weggevallen.

Bij de beschrijving van het statement ASC2BAS dient toegevoegd te worden dat regels die alleen bestaan uit een regelnummer en

spaties vermeden moeten worden. Tijdens het runnen van een programma kunnen er problemen ontstaan bij het gebruik van labels.
Mocht u toch lege regels willen toevoegen, plaats dan achter het regelnummer een ; of een ander teken. Na het converteren kunnen deze regels m.b.v. P-Charme snel gewijzigd worden.

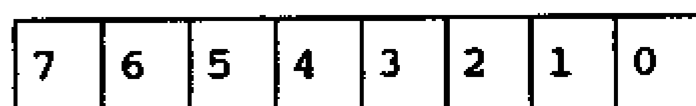
```
Voorbeeld:      in te voeren regel: 1340 ~
                na converteren:     /~/ /
                resultaat:          1340
```

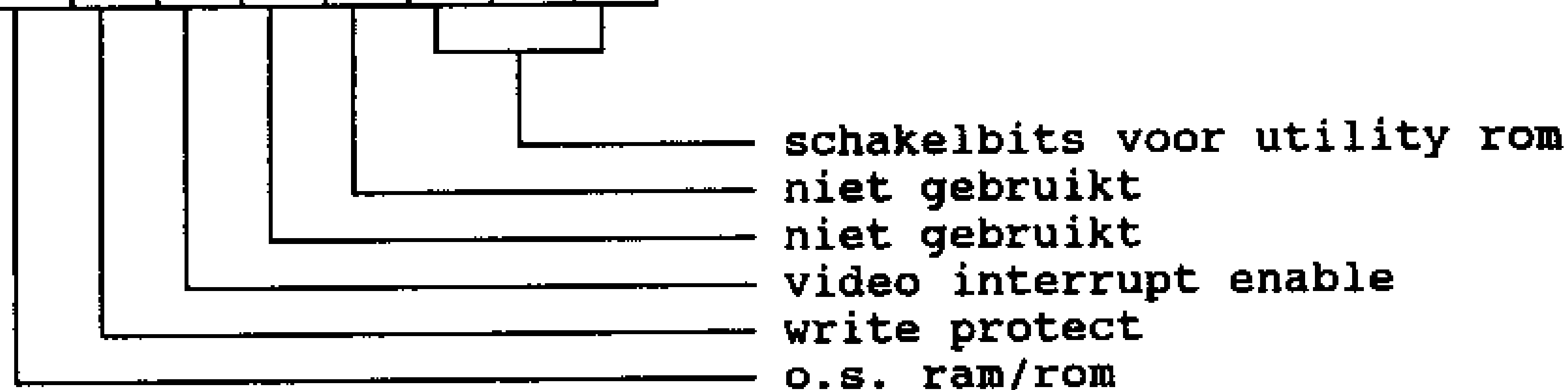
Verder zijn in de beschrijving van de PCBOX twee statements vergeten:
Om een printer te kiezen kan men geven
     LPT =
of   PRN =
gevolgd door het printernummer. Alle uitvoer na $2 wordt dan naar die printer gestuurd.

Na het drukken van de handleiding is nog een belangrijke wijziging aangebracht in het operating systeem. De keyboard scan routine op #FE71 wordt zo goed als mogelijk gesimuleerd. Hetzelfde geldt voor de leesroutine op #FE94. Het wordt nog steeds ten zeerste afgeraden om deze routine aan te roepen omdat deze niet via de leesvector loopt!

Al vanaf het eerste artikel over de Atom in PC werd er geschreven en gesproken over ram in plaats van eprom op het CDEF-blok en write protect. Maar in de handleiding is er nergens één letter over geschreven hoe dat in of uitgeschakeld wordt. Bij deze dus:

De bits van de latch op #B888 hebben de volgende betekenis:






De meest interessante bits voor de programmeur zijn bit 7 en bit 6. De schakelbits worden geheel verzorgd door Branquart en de video interrupt enable wordt onder andere ingesteld bij het kiezen van een grafische mode m.b.v. het CLEAR statement.

Als een van bovenstaande bits gewijzigd wordt dan moet ervoor gezorgd worden dat de overige bits van de latch niet verande-

ren. Daartoe hebben we twee schaduwbytes. Een schaduwbyte is geplaatst in #4FF. Daarin bevindt zich het nummer van de voorgeschakelde utility bank op het A-blok. Bit 6 van dit schaduwbyte wordt door Branquart gebruikt om een box vast te zetten (lock toolbox). De bits 3, 4, 5 en 7 dienen '0' te zijn. Het tweede schakelbyte bevindt zich in #4FD. Daarin is de status opgeslagen van de hoogste vijf bits van de latch. De laagste drie bits dienen '0' te zijn. Om nu een van deze vijf bits te wijzigen leest men de waarde van het tweede schaduwbyte. Wijzig daarin de betreffende bits en schrijf de waarde terug. Lees vervolgens het eerste schaduwbyte; reset bit 6 en voer een OR funktie uit met het tweede schaduwbyte. Het resultaat daarvan mag dan naar de latch geschreven worden.

In basic ziet dat er dan zo uit:

```
10 REM WRITE PROTECT UIT ZETTEN
20 ?#4FD=?#4FD & #BF      ; REM WIS BIT 6
30 ?#B888=(?#4FF & #BF) | ?#4FD
40 REM WRITE PROTECT STAAT NU UIT
```

In assembler gaat dat als volgt:

```
LDA #4FD        \ LAADT SCHADUWBYTE 2
AND @#BF        \ RESET BIT 6, WRITE PROTECT OFF
STA #4FD        \ SCHRIJF RESULTAAT TERUG
LDA #4FF        \ LAADT SCHADUWBYTE 1
AND @#BF        \ WIS LOCK BIT
ORA #4FD        \ VOEG BEIDE SCHADUWBYTES SAMEN
STA #B888       \ SCHRIJF NAAR DE LATCH
```

Dit is natuurlijk te omslachtig in veel gevallen. Vanuit basic is write protect aan en uit te zetten met het WP statement. Het overschakelen tussen operating systeem in RAM en ROM, dat op soortgelijke wijze gebeurt, moet met enige voorzichtigheid gebeuren. Zeker als er in het RAM geen operating systeem geladen is. Vanuit BASIC is er dus weinig toepassing voor deze mogelijkheid. Assembler programma's kunnen er eventueel data of routines plaatsen en daarvoor steeds schakelen tussen ram en rom.

Om het verwarrend te maken kunnen basicprogramma's wel machinetaal routines aanroepen die op hun beurt gebruik maken van die extra ram en daarna weer rom voorschakelen.

**Nieuwe uitbreidingen in versie 2.xx**

De uitbreidingen worden verdeeld in drie groepen:
a) nieuwe control codes
b) nieuwe pc commando's
c) communicatie met andere programma's

In enkele bestaande routines uit versie 1.xx zijn wijzigingen aangebracht maar deze zijn niet merkbaar voor de gebruiker. Alle commando's zijn wat aanroep betreft niet gewijzigd.

**Nieuwe control codes**

De nieuwe control codes lijken veel op de control codes van de BBC en Electron. Er zijn vier manieren om de nieuwe control codes aan te roepen. De eerste is via het toetsenbord, CTRL+letter. De tweede is via het PRINT statement; de derde is met het VCC statement (zie nieuwe toolbox) en de vierde is met een overeenkomstig statement (ook in de nieuwe toolbox).

Het probleem met CTRL-toets en PRINT statement is dat een $0 als twee nullen naar de PC gestuurd moet worden. Dat geeft problemen als er een 0 als parameter gegeven wordt. In dat geval wordt er een 0 te veel overgestuurd. Verder wordt de semafoor niet getest en gezet. Er mag bij het oversturen van de parameters geen interrupt optreden waarbij gecommuniceerd wordt met de PC. Opletten dus!

Het VCC statement lost deze problemen grotendeels op. De semafoor wordt getest en gezet; daarna worden de codes overgestuurd, waarbij een 0 slechts éénmaal wordt verzonden. Na de laatste code wordt de semafoor gewist. Er kunnen alleen problemen onstaan als er tijdens de uitvoering van het VCC statement een foutmelding ontstaat (bijvoorbeeld een foutieve parameter opgegeven). Er wordt in dat geval geprobeerd om een foutmelding te geven waarbij de semafoor getest wordt, maar deze is gezet door het VCC statement. Het systeem hangt.

De laatste manier levert helemaal geen problemen op; de statements om de nieuwe controlcodes uit te voeren worden beschreven in het hoofdstuk 'Een nieuwe toolbox'.

| | | |
|---|---|---|
| CTRL A | $1 | Het hierna volgende teken wordt naar de printer gestuurd indien deze aangezet was met $2. |
| CTRL D | $4 | Selecteert de schermpagina die op de monitor verschijnt. De waarde van deze pagina wordt gegeven na $4. |
| CTRL E | $5 | Selecteert de schermpagina waar alle schrijfopdrachten plaats vinden. Deze pagina hoeft niet in beeld te zijn. |
| CTRL Q | $17 | Verandert de kleur. De nieuwe kleur (attribuut) wordt opgegeven na $17. |
| CTRL V | $22 | Verandert de schermmode van de PC. Let erop dat bij deze vorm van mode selectie geen rekening |

gehouden wordt met de plotvector en Atom modi.

CTRL W $23 Definieert een grafisch karakter voor de CGA grafische modi. Na $23 volgt het te definieren karakter (>= 128). Daarna volgen 8 bytes die het bitpatroon van het karakter bevatten.

CTRL X $24 Definieert een grafisch venster in extended video mode. De hoeken linksonder en rechtsboven worden als vier maal 2 bytes overgestuurd, low byte eerst. De volgorde is XLO, YLO, XRB, YRB.

CTRL Z $26 Heft alle gedefinieerde vensters op.

$28 Definieert een tekstvenster. Gelijk aan het statement TWINDOW in de PCBOX.

$29 Verplaatst de grafische oorsprong. Na de opdracht $29$x$y komt het punt (0,0) te liggen op de positie x,y. Let er wel op dat x en y ieder twee bytes groot zijn.

$31 Verplaatst de cursor in tekstmode en extended video mode naar de opgegeven positie.

Deze control codes werken alleen in de tekstmode of in extended video mode. Ze dienen vermeden te worden in Atomvdu of grmod.

**Nieuwe PC commando's**

Onderstaande PC commando's zijn toegevoegd in versie 2.xx:

**#47 Teken naar printer sturen**
Invoer: teken (1 byte)
Uitvoer: geen
Het gelezen teken wordt naar de printer gestuurd indien deze aangezet is met $2 of ^B. Als de printer niet aangezet is dan voert dit commando niets uit. In het geval dat de printer wel aangezet is maar de printer is niet on-line wordt gewacht totdat de printer on-line komt.

**#48 CGA grafisch teken definiëren**
Invoer: ascii waarde van teken (1 byte)
bitpatroon van teken (8 bytes)
Uitvoer: geen
In de grafische CGA mode is het mogelijk om zelf de tekens 128 t/m 255 te definiëren (iets dat MS-DOS doet met het commando GRAFTABL). Deze mogelijkheid is nu ook beschikbaar voor de Atom.
Nadat de ascii waarde van het te definiëren teken is

overgezonden worden de volgende acht bytes opgeslagen als bitpatroon.
De nieuwe tekens worden in hetzelfde gebied opgeslagen als het font voor GTEXT en Atomvdu; het is dus niet mogelijk om deze drie fonts gelijktijdig te gebruiken of de fonts moeten om en om geladen worden.

*Het is al aan te voelen dat een komplete karakterset geladen kan worden. Om een nieuwe karakterset te laden kan het bekende FONT statement gebruikt worden.*

**#49 Horizontaal vullen**

Invoer: coördinaten beginpunt (2 words)
grenskleur (1 byte)
Uitvoer: geen
Vanaf het opgegeven punt wordt de beeldlijn naar links en naar rechts gekleurd in de aktuele kleur. Dit kleuren gaat door totdat een pixel ontmoet wordt dat gelijk is aan de grenskleur.
Bij het kleuren wordt rekening gehouden met een eventueel grafisch venster en een verschoven oorsprong. Voorwaarde is wel dat het beginpunt in het grafisch venster ligt.

**#4A Vertikaal vullen**

Invoer: coördinaten beginpunt (2 words)
grenskleur (1 byte)
Uitvoer: geen
Vanaf het opgegeven punt wordt de beeldlijn naar boven en naar beneden gekleurd in de aktuele kleur. Dit kleuren gaat door totdat een pixel ontmoet wordt dat gelijk is aan de grenskleur.
Bij het kleuren wordt rekening gehouden met een eventueel grafisch venster en een verschoven oorsprong. Voorwaarde is wel dat het beginpunt in het grafisch venster ligt.

**#4B Aanroepen INT 61h**

Invoer: afhankelijk van aangeroepen programma
Uitvoer: afhankelijk van aangeroepen programma
Om het Atom terminal programma zelf uit te breiden kan op de PC een resident programma gestart worden dat geaktiveerd wordt bij een oproep van INT 61h. Parameters en resultaten kunnen afhankelijk van het opgeroepen programma uitgewisseld worden.
Zie het hoofdstuk "communicatie met andere programma's" voor meer uitleg.

**#4C Definiëer een grafisch venster**

Invoer: X-coördinaat linker benedenhoek (1 word)
X-coördinaat linker benedenhoek (1 word)
Y-coördinaat rechter bovenhoek (1 word)
Y-coördinaat rechter bovenhoek (1 word)
Uitvoer: geen
Voor de Atom grafische mode bood het Gagsrom al mogelijk-

heden voor grafische vensters. Voor de extended video mode kan dit commando gebruikt worden om een grafisch venster vast te leggen. Dat houdt in dat alleen binnen dat venster getekend kan worden.
Na het gebruiken van een CLEAR opdracht wordt het venster automatisch opgeheven.

**#4D Grafisch venster opheffen**
Invoer: geen
Uitvoer: geen
Hiermee wordt een grafisch venster opgeheven.

**#4E Grafische oorsprong verschuiven**
Invoer: coördinaten van nieuwe oorsprong (2 words)
Uitvoer: geen
Het grafische nulpunt ligt normaal in de linker bovenhoek van het scherm. Met dit commando kan het punt (0,0) verschoven worden naar een willekeurige plaats op het scherm.
Bij het verschuiven van de grafische oorsprong wordt geen rekening gehouden met een eerder verschoven oorsprong en een grafisch venster. De nieuwe positie is dus altijd ten opzichte van de linker bovenhoek van het scherm.
Na een CLEAR opdracht bevindt de oorsprong zich altijd in de linker bovenhoek.

**#4F Selecteer schermpagina voor monitor**
Invoer: schermpagina nummer (1 byte)
Uitvoer: geen
Bij gebruik van een tekstmode kan gebruik gemaakt worden van verschillende schermpagina's. Er zijn in feite twee mogelijkheden bij het werken met verschillende pagina's, te weten: werkpagina is zichtbaar op scherm of de werkpagina en de getoonde pagina zijn verschillend. Met het commando #4F kan de pagina gekozen worden die op de monitor getoond wordt.
Zie voor meer informatie het hoofdstuk "Schermpagina's".

**#50 Selecteer schermpagina voor bewerkingen**
Invoer: schermpagina nummer (1 byte)
Uitvoer: geen
Bij gebruik van een tekstmode kan gebruik gemaakt worden van verschillende schermpagina's. Er zijn in feite twee mogelijkheden bij het werken met verschillende pagina's, te weten: werkpagina is zichtbaar op scherm of de werkpagina en de getoonde pagina zijn verschillend. Met het commando #50 kan de pagina gekozen worden waar de schrijfopdrachten naar toe gestuurd worden.
Zie voor meer informatie het hoofdstuk "Schermpagina's".

**#51 Zet cursor lijnen**
Invoer: bovenste cursorlijn (1 byte)
onderste cursorlijn (1 byte)

Uitvoer: geen
De vorm van de cursor kan ingesteld worden met dit commando. De cursorlijnen worden van boven naar beneden geteld. De eerste parameter zal dus altijd groter zijn dan de tweede. Bij ongeldige waarden verdwijnt de cursor van het scherm. Het aantal lijnen verschilt per videokaart.

**#52 Gereserveerd**

**#53 Gereserveerd**

**#54 Gereserveerd**

**#55 Selecteer user font voor EGA/VGA tekstmode**
Invoer: geen
Uitvoer: geen
Dit commando laadt het geladen font in het fontram van de EGA of VGA kaart. Hiermee kunnen de karakters in tekstmode gewijzigd worden.

**#56 Selecteer EGA 8*14 ROM font**
Invoer: geen
Uitvoer: geen
Hiermee wordt het standaard EGA 8*14 karakterset in de fontram van de EGA of VGA kaart geladen. Automatisch wordt omgeschakeld naar 25 regels per scherm. Het kan nodig zijn om het tekstwindow te herdefiniëren.

**#57 Selecteer EGA 8*8 ROM font**
Invoer: geen
Uitvoer: geen
Hiermee wordt het standaard EGA 8*8 karakterset in de fontram van de EGA of VGA kaart geladen. Automatisch wordt omgeschakeld naar 43 regels per scherm voor EGA en 50 regels per scherm voor VGA. Het tekstwindow zal aangepast moeten worden.

**Schermpagina's**

In het nieuwe terminalprogramma worden de verschillende schermpagina's in tekstmode ook ondersteund. Voor grafische toepassingen blijft steeds één pagina beschikbaar. De meeste videokaarten bieden voor 40 kolomsschermen tot 8 pagina's en voor 80 kolommen tot 4 pagina's.

Het is (met enige beperking) mogelijk om een beeldpagina op de monitor te plaatsen en daarna in een andere pagina te schrijven. De beperkingen hebben betrekking op het scrollen en het wissen van een scherm of venster. Deze beperkingen worden veroorzaakt door het BIOS van de PC. (Iemand moet toch de schuld krijgen).

Als het CLS commando gebruikt wordt, wordt alleen de beeldpagina die op de monitor verschijnt gewist en blijft de beeldpagina ongewijzigd. Het CLS commando heeft een verbeterde werking ten opzichte van versie 1.xx -> het gehele scherm wordt gevuld met het geldende attribuut. Het CLW statement werkte ook al zo. CLS reset het tekstvenster. Met het CLEAR x statement kunnen alle pagina's tegelijk gewist worden.

Ook DOS heeft wat moeite met het werken met verschillende pagina's. Foutmeldingen en de uitvoer van *CAT worden op het scherm getoond via functies van DOS INT 21h. Hierbij wordt geen rekening gehouden met vensters en ook niet met schermpagina's. Dat heeft als gevolg dat de geproduceerde uitvoer altijd op de monitor verschijnt, ook al is een andere pagina geselecteerd voor uitvoer.

Een aardige toepassing van schermpagina's is het oproepen van een HELP-scherm waarbij het programmascherm niet verloren gaat. Een andere toepassing is het laten knipperen van tekst in verschillende kleuren. Plaats daartoe de tekst in verschillende kleuren in verschillende beeldpagina's maar steeds op dezelfde plaats. Laat daarna in een rustig tempo de schermen elkaar afwisselen. Het resultaat is dat de tekst in verschillende kleuren knippert.

```
10 REM KNIPPERENDE TEKSTEN
20 CLEAR 7
30 PRINT $5$1; ATTRIB #0E; PRINT "DIT IS PAS KNIPPEREN"
40 PRINT $5$2; ATTRIB #08; PRINT "DIT IS PAS KNIPPEREN"
50 PRINT $5$3; ATTRIB #0A; PRINT "DIT IS PAS KNIPPEREN"
60 DO FOR S=1 TO 3
70 PRINT $4$S ; PAUSE 9
80 NEXT S
90 UNTIL 0
```

Verder is het mogelijk om de copy-toets te gebruiken in alle schermen. De copy toets werkt overigens in de geselecteerde werkpagina.

**Communicatie met andere programma's**

Er zijn situaties denkbaar waarbij het wenselijk is als er vanuit een ander programma dat in de PC draait, met de Atom gecommuniceerd kan worden. In de beschrijving van de nieuwe PC-commando's is reeds een mogelijkheid gegeven via INT 61h. Een andere mogelijkheid wordt geboden via INT 60h. Hierbij worden de volgende drie functie's herkend:

**INT 60h - functie 0** (AH=00h)
De waarde in AL wordt van de PC naar de Atom gestuurd.

**INT 60h - functie 1** (AH=01h)
Er wordt een byte van de Atom gelezen en in AL geplaatst.

**INT 60h - functie 2** (AH=02h)
Het statusregister van de communicatie PPI wordt geretourneerd in AL. Hiermee kan eventueel getest worden of een teken gereed staat. In DX wordt het basisadres van de 8255 PPI teruggegeven.

Met uitzondering van het statusregister worden de niet vermelde registers niet gewijzigd.

INT 60h geeft een programma in de PC de mogelijkheid om een byte naar de Atom te sturen of van de Atom te lezen. Twee voorbeelden van toepassingen zijn:

- Er dienen een groot aantal bytes zo snel mogelijk van de Atom naar een I/O poort gezonden te worden, bijvoorbeeld in blokken van 1024 bytes. Dat kan gebeuren door steeds commando #24 te gebruiken. Echter daarbij worden voor elk byte vijf bytes gestuurd van de Atom naar de PC.
  De Atom kan ook eerst een programma starten op de PC en vervolgens die 1024 bytes naar de PC sturen. Het Atomprogramma kan er zo uitzien:

```
10 ESCAPE = OFF
20 *PC BYTES2IO
30 FOR X=1 TO 1024
40 A=X?#4000          ; REM LEES DATA
50 LINK #FFC8         ; REM STUUR NAAR PC
60 NEXT X
70 ESCAPE = ON
80 END
```

  Het is belangrijk dat in dit geval de ESCAPE toets uitgezet wordt. Na ieder statement test de BASIC interpreter of de escape toets is ingedrukt. Hiertoe wordt een commando (#01) naar de PC gestuurd. Aangezien het gestartte programma BYTES2IO daar niet op rekent moet dat vermeden worden. Regel 10 zorgt daarvoor. Er moet tevens rekening mee gehouden worden dat de semafoor bediend wordt of men moet er zeker van zijn dat er geen interrupt optreedt waarbij met de PC gecommuniceerd kan worden.
  De semafoor kan in BASIC gezet worden met A=11;LINK #FFC2 en met A=10;LINK #FFC2 wordt de semafoor weer gewist.

Het programma BYTES2IO op de PC kan er in grote lijnen zo uitzien:

```
      MOV  AH,01H         ; LAADT FUNCTIENR VOOR INT 60
      MOV  DX,ioadres     ; ZET I/O ADRES IN DX REGISTER
      MOV  CX,1024        ; LAADT TELLER
lus:  INT  60H            ; LEES BYTE VAN DE ATOM
      OUT  DX,AL          ; STUUR BYTE NAAR I/O POORT
      LOOP lus            ; CX=CX-1;ALS CX<>0 GA NAAR lus
      RET                 ; EINDE PROGRAMMA
```

Er worden bij de laatste methode alleen maar de 1024 databytes overgezonden, terwijl bij de eerste methode (d.m.v. commando #24) 5120 bytes overgezonden worden.

Een ander voorbeeld is een wiskundig probleem. Dit voorbeeld staat ook op de diskette met de update.

- Er moet een moeilijke en langdurige berekening worden uitgevoerd:
  Bereken   (7-2) + (8-7)
  Aangezien we twee processoren (een 80x86 en een 6502) in het systeem hebben zitten kunnen we ditmaal de Atom als rekenkundige insteekkaart voor de PC gebruiken.
  Op de Atom draait onderstaand programma:

```
10 ESCAPE = OFF
15 *COMMAND
20 ZERO          ; REM ALLE VARIABELEN 0 MAKEN
30 USR #FFC5     ; REM LEES EERSTE GETAL
40 B=A           ; REM BEWAAR DIT IN DE VARIABELE B
50 USR #FFC5     ; REM LEES TWEEDE GETAL
60 A=B-A         ; REM VOER BEREKENING UIT
70 LINK #FFC8    ; REM STUUR RESULTAAT NAAR PC
80 ESCAPE = ON
90 END
```

  Na run verschijnt de MS-DOS prompt weer bij de PC. Start dan het onderstaande PC-programma (gecompileerd in Turbo C: MOEILIJK.EXE)

```
#include <stdio.h>
#include <dos.h>

main()
{
int pc_uitkomst, atom_uitkomst, totaal;
union REGS regs;

printf ("Een voorbeeld van parallelprocessing,\n");
printf ("twee processoren berekenen een moeilijk");
printf (" wiskundig probleem ... ");
```

```
/* Stuur data naar de Atom */

regs.x.ax = 0x0008;
int86(0x60, &regs, &regs);
regs.x.ax = 0x0007;
int86(0x60, &regs, &regs);

/* De Atom heeft nu de data ontvangen en
   kan beginnen met het uitrekenen van 8-7.
   De PC berekent op datzelfde moment 7-2. */

pc_uitkomst = 7 - 2;

/* Lees nu de uitkomst van de Atom */

regs.x.ax = 0x0100;
int86(0x60, &regs, &regs);
atom_uitkomst = regs.x.ax & 0xFF;

/*  Bereken de totale som en druk het resultaat
     af */

totaal = pc_uitkomst + atom_uitkomst;

printf ("\n\n(7-2) + (8-7) = %ld\n\n", totaal);
}
```

Ofschoon bovenstaand voorbeeld erg omslachtig is om de opgegeven som te berekenen is dit een mooi voorbeeld van parallel processing. Overigens werken de Atom en PC toch al veel samen op basis van parallelprocessing, bijvoorbeeld het tekenen van een lijn: de Atom berekent welk punt gezet moet worden, geeft dat door aan de PC. Terwijl de PC het punt op het scherm zet berekent de Atom al het volgende punt.

## PC UTILITY BOX

Om van de nieuwe mogelijkheden van het terminal programma gemakkelijk gebruik te maken in Basic is er weer een nieuwe toolbox bij gekomen. Deze toolbox heeft de naam PC Utility Box; op de diskette te vinden onder de naam PCUTIL.ROM.

De statements in deze box zijn in te delen in twee categorieen, enerzijds statements die commando's naar de PC sturen om de gewenste functie uit te voeren, anderzijds zijn er een aantal statements opgenomen die snel en eenvoudig een PC programma opstarten.

Vanaf de volgende bladzijde staat een overzicht van de statements:

**BASIC start PC basic interpreter/compiler**
Syntax: BASIC
Dit statement is een voorbeeld van de categorie statements die een PC programma starten. Vanuit de Atom wordt een Basic interpreter/compiler op de PC gestart. Het pad en de naam van deze Basic is opgeslagen vanaf adres #AFA0, als ascii string van maximaal 31 karakters.

**CHAIN laadt en start Atom basicprogramma**
Syntax: CHAIN "filespec"
Het CHAIN statement laadt een basicprogramma in de actuele tekstpagina en start het programma. Het executieadres maakt in dit geval niets uit omdat het programma altijd gestart wordt vanaf #F144. De filespecificatie mag een drive en directory bevatten maar geen wildcards * en ?.

**EDIT start editor**
Syntax: EDIT
In de PCBOX is het statement ASC2BAS opgenomen. Daarmee kan een ascii file geladen en geconverteerd worden naar een Atom basicprogramma. Het statement EDIT is opgenomen om snel een ascii teksteditor op de PC te starten. Het pad en de filenaam van deze editor zijn opgeslagen als ascii string van maximaal 31 karakters vanaf adres #AF20.

**ED64 start tekstverwerker**
Syntax: ED64
Dit oude bekende commando start een tekstverwerker op de PC. Zo kunt u snel een tekstverwerker starten om bij het zojuist geschreven programma een artikel voor Atom Nieuws te schrijven. Het pad en de filenaam van deze tekstverwerker zijn opgeslagen als ascii string van maximaal 31 karakters vanaf adres #AF00.

**FILECNV fileheader toevoegen of weghalen**
Syntax: FILECNV filespec1 filespec2 H+ [start exec]
FILECNV filespec1 filespec2 H-
De files voor de Atom bevatten altijd een header van 16 bytes aan het begin van de file. Met het statement FILECNV kan deze header toegevoegd of weggehaald worden, bijvoorbeeld als er files van een "echte" Atom en de Atom in PC worden uitgewisseld.
Filespec1 bevat het pad en filenaam van de oorspronkelijke file; deze file wordt niet veranderd. Filespec2 is het pad en de filenaam van de nieuwe file waar de header is toegevoegd of weggehaald. H+ geeft aan dat de header wordt toegevoegd; H- betekent header weghalen. Het start en executie adres mogen weggelaten worden. In dat geval worden de waarden #2900 en #F144 gebruikt. Geef deze adressen met # teken en zonder komma tussen de parameters.
Let erop dat de originele file en de nieuwe file niet hetzelfde zijn; dan gaan alle data verloren!

**GWINDOW definieëren grafisch venster**
Syntax: GWINDOW xlo,ylo,xrb,yrb
Dit statement werkt alleen in de extended videomodi. Er wordt een grafisch venster gecreeërd waarin getekend mag worden. De vier parameters geven de X en Y coördinaten aan van de linker benedenhoek en de rechter bovenhoek. Deze coördinaten gelden altijd ten opzichte van de linker bovenhoek van het beeldscherm.

**GWOFF grafisch venster opheffen**
Syntax: GWOFF
Het grafische venster wordt gereset zodat weer over het gehele scherm getekend mag worden.

**HELP on-line helpfunctie**
Syntax: HELP [trefwoord]
Hoe was de syntax ook alweer van het HLINE commando? Hoeveel parameters krijgt het UDS statements mee? Met welk PC-commando wordt de printerstatus opgevraagd?
Inmiddels zijn er voor de Atom zovele toolboxen en nog meer extra statements toegevoegd dat het bijna onmogelijk is om nog een eenvoudig overzicht te houden. Bovenstaande vragen kunnen nu snel beantwoord worden met het HELP statement. Op het scherm verschijnt razendsnel voldoende maar toch beknopte uitleg.
HELP gebruikt twee datafiles, ATOMHELP.IDX en ATOMHELP.TXT; het pad en de naam van deze datafiles zijn opgeslagen als ascii string vanaf adres #AFC0 resp. #AFE0.
Bij gebruik van HELP wordt eerst het trefwoord opgezocht in de indexfile. Als het daar gevonden is, wordt een pointer gelezen naar het begin van de helptekst in de file ATOMHELP.TXT. Die tekst wordt vervolgens op het scherm afgebeeld.

Het wijzigen of toevoegen van de helptekst kan met een gewone tekst editor. Plaats de trefwoorden aan het begin van de regel, voorafgegaan door een golfje ~. Achter het trefwoord moet minstens één spatie staan. Het laatste trefwoord moet ~end zijn. In de eerste regel staan zestien apestaartjes (@); deze niet verwijderen want zij vormen de header van de tekstfile.
Na het wijzigen of toevoegen aan de helptekst moet de indexfile opnieuw aangemaakt worden. Verwijder de oude ATOMHELP.IDX file en laat het programma HELPMKR even lopen. Doe dit bij voorkeur op een ramdisk (in verband met snelheid).

**HFILL horizonale lijnen kleuren**
Syntax: HFILL x,y1 TO y2,g
Met HFILL kunnen horizontale lijnen gekleurd worden. Het kleuren begint bij x,y1 en kleurt naar links en rechts totdat een pixel wordt aangetroffen die de grenskleur g heeft. Vervolgens wordt de y-coördinaat verhoogd en wordt de volgende lijn ingekleurd. HFILL werkt alleen in extended videomodi en houdt rekening met een grafisch venster en een verschoven

oorsprong.

**POWERDOWN snel geheugeninhoud op disk bewaren**
Syntax: POWERDOWN
Dit statement is bedoeld om snel de geheugeninhoud van #300 tot #A000 op disk te bewaren als de computer uit wordt gezet. Na het opnieuw opstarten kan de geheugeninhoud hersteld worden met *LOAD MEMORY.PD

**MODEM start communicatie software**
Syntax: MODEM
Dit statement start een communicatie software pakket op de PC. Het pad en de filenaam van dit pakket is als ascii string opgeslagen vanaf adres #AF40.

**ORIGIN verplaatsen grafische oorsprong**
Syntax: ORIGIN x,y
De grafische oorsprong (het punt 0,0) wordt voor extended videomodi verschoven naar het punt x,y; dit punt is opgegeven ten opzichte van de linker bovenhoek van het beeldscherm.

**OSCLI aanroep *-commando**
Syntax: OSCLI string,$13
In PC-DOS is het niet toegestaan om variabelen te gebruiken bij de *-commando's. Met OSCLI is dat gemis te omzeilen. De op te geven string kan bestaan uit tekst tussen aanhalingstekens en (string)variabelen. Let erop dat dit statement altijd wordt afgesloten met een $13 !

**PCC printer control code**
Syntax: PCC p1,p2,p3,.....
Als de printer aangezet is met $2 stuurt PCC alle opgegeven parameters uitsluitend naar de geselecteerde printer. Ook de codes $2, $3 en de inhoud van #FE worden naar de printer gestuurd. De printer kan met PCC niet aan of uitgezet worden.

**PCTOOLS start PC-Tools**
Syntax: PCTOOLS
Dit statement start het gelijknamige hulpprogramma. Het pad en filenaam zijn opgeslagen als ascii string van maximaal 31 karakters vanaf #AF60.

**SCREEN selecteren beeldpagina's**
Syntax: SCREEN s
De meeste videokaarten kunnen overweg met meerdere schermpagina's. De Atom kan inmiddels ook overweg met meerdere pagina's in tekstmode. Met SCREEN kunnen pagina's opgeroepen worden. Als s tussen 0 en 7 ligt wordt dat de pagina die op de monitor verschijnt en waarin de bewerkingen plaatsvinden. Een s tussen #40 en #47 selecteert alleen de pagina waarin bewerkingen plaatsvinden en s tussen #80 en #87 selecteert alleen de pagina die op de monitor verschijnt.

**TEKENPRG start tekenprogramma**
Syntax: TEKENPRG
Dit statement start een tekenpakket op de PC, waarvan het pad en de naam als ascii string zijn opgeslagen vanaf #AF80.

**TXMOD omschakelen naar tekstmode**
Syntax: TXMOD [optie]
Dit statement vervangt het gelijknamige statement uit de PCBOX (vanaf versie 1.03 komt dat statement daar niet meer in voor) en kan nu ook gebruikt worden met de volgende optie's voor EGA en VGA kaarten:

**TXMOD /8**
Er wordt naar tekstmode geschakeld waarbij het EGA 8*8 karakterfont wordt geladen; automatisch wordt overgeschakeld naar 43 regels per scherm voor EGA kaarten of 50 regels per scherm voor VGA kaarten. Het tekstvenster wordt automatisch goed gezet mits de juiste video-optie is opgegeven bij het starten van het terminal programma.

**TXMOD /14**
De tekstmode wordt ingeschakeld waarbij het EGA 8*14 karakterfont wordt geladen; automatisch wordt overgeschakeld naar 25 regels per scherm voor EGA kaarten of 28 regels per scherm voor VGA kaarten. Het tekstvenster wordt automatisch goed gezet mits de juiste video-optie is opgegeven bij het starten van het terminal programma.

**TXMOD /F pad+fontnaam**
De tekstmode wordt ingeschakeld, het opgegeven font wordt van disk geladen en wordt geaktiveerd.

**TXMOD /C**
Het in de buffer geladen font (laden m.b.v. het statement FONT) wordt gekopiëerd naar fontram van de EGA of VGA kaart waarbij de op het scherm aanwezige tekst wordt aangepast aan het nieuwe font. De overige instellingen blijven ongewijzigd.

Het TXMOD statement is geheel compatible met het oude TXMOD statement uit de PCBOX en kan dus ook aangeroepen worden met CGA of Hercules videokaarten. Uiteraard zullen de gegeven opties geen werking hebben bij deze kaarten.

Een andere wijziging van het TXMOD statement is dat de NMI niet meer geblokkeerd wordt na het aanroepen van dit statement; dit in verband met I/O emulatie (zie verderop in de tekst).

**UTIL overzicht statements PC Utility Box**
Syntax: UTIL
Het scherm wordt gewist en er verschijnt een overzicht van alle statements uit deze toolbox.

**VCC video control code**
Syntax: VCC p1,p2,p3,....
VCC stuurt alle opgegeven parameters direct door naar de PC. De semafoor wordt gedurende de uitvoering van dit statement gezet zodat het oversturen niet onderbroken kan worden. Een $0 wordt als slechts één 0 overgestuurd. Dit statement is goed te gebruiken om via control codes bepaalde commando's uit te voeren.

**VDU video control code**
Syntax: VDU p1,p2,p3,...
Dit statement is gelijk aan het VCC statement.

**VFILL vertikale lijnen kleuren**
Syntax: VFILL x1 TO x2,y,g
Met VFILL kunnen vertikale lijnen gekleurd worden. Het kleuren begint bij x1,y en kleurt omlaag en omhoog totdat een pixel wordt aangetroffen die de grenskleur g heeft. Vervolgens wordt de x-coördinaat verhoogd en wordt de volgende lijn ingekleurd. VFILL werkt alleen in extended videomodi en houdt rekening met een grafisch venster en een verschoven oorsprong.

**Het gebruik van een Herculeskaart**

In de eerste versies van het terminalprogramma worden de mogelijkheden van de Herculeskaart helemaal niet ondersteund. In de nieuwe versie is enigzins rekening gehouden met mogelijkheden van de Herculeskaart, ofschoon er wel twee hulpprogramma's noodzakelijk zijn.

Het eerste programma is MSHERC.COM dat de videofuncties van INT 10h geschikt maakt voor de Herculeskaart. Dit residente programma moet geïnstalleerd worden voordat de Atom gestart wordt. De Atom schermemulatie (CLEAR 0 t/m 4) wordt daarmee nog niet ondersteund! Wel is het mogelijk om via de plotvector te tekenen in de Hercules resolutie 720*348. Dit is de extended videomode CLEAR 13.

Een tweede hulpprogramma is HGCIBM.COM; een bekende CGA emulator voor de herculeskaart. Als dit programma geïnstalleerd is kunnen de CGA modi gebruikt worden en dus ook de Atom schermemulatie. HGCIBM kan eventueel vanuit de Atom gestart worden d.m.v. *PC HGCIBM gevolgd door de nodige parameters. Na afloop kan dit programma weer gedeïnstalleerd worden waardoor de Herculeskaart weer beschikbaar komt. Beide programma's zijn op de update disk geplaatst.

Als u gebruik maakt van de Herculeskaart wordt het aanbevolen om bij het starten van ATOM.COM de optie /V=HGC te geven; hiermee wordt het voor de Atom mogelijk om ook de Herculeskaart te detecteren en eventueel kan in een programma de

aanroep van HGCIBM automatisch verzorgd worden. Verder zorgt deze optie ervoor dat de cursor niet midden op de regel zweeft.

**Wijzigingen PC-BOX**

Het TXMOD statement is vanwege veranderingen en uitbreidingen verhuisd naar de PC Utility Box.

Een andere wijziging in de PCBOX is een verbetering van het statement ATOMVDU. In de praktijk leverde het oude statement vaker problemen op; meestal hing alleen de Atom, soms zelf de hele PC. Dit statement is nu verbeterd en het is niet meer nodig om eerst CLEAR 0 in te schakelen.

> Voordat ATOMVDU gebruikt kan worden moet de karakterset ATOMVDU.FNT geladen zijn.

De nieuwe PCBOX heeft 1.03 als versienummer.

**Correctie PC-DOS**

Ook PC-DOS bleek niet geheel vrij van fouten. In sommige gevallen bleek er iets fout te gaan als een filenaam vooraf gegaan werd door een pad. Meestal leidde dat tot een foutmelding en zeer af en toe gingen er ongewenst wat files verloren.

Een minder opvallende fout was dat het *SAVE commando files kon wegschrijven met een maximale lengte van 32 KB. Als er een langer bereik werd opgegeven dan werden er slechts 256 bytes op de disk gezet.

In PC-DOS v1.26 zijn deze fouten verholpen.

Tot zover de aanvullingen op de handleiding en de nieuwe uitbreidingen. Als er vragen of problemen zijn, neem dan contact met mij op. Ik zal dan het probleem zo snel mogelijk opgelossen. Eventuele wijzigingen in de software kunnen dan eventueel via een modem worden overgestuurd.

De nieuwe software wordt op de gebruikelijke wijze verspreid en u kunt eventueel na afspraak met een modem de software downloaden.

**Atom.Com en andere operating-systemen.**

Ofschoon het terminalprogramma geschreven is voor een MS-DOS

omgeving draait het ook probleemloos in een DOS-venster onder Windows 3.1. Een geschikte ATOM.PIF en een icoon zijn bijgevoegd op de diskette met de nieuwe software. Het is uiteraard mogelijk om de Atom op de achtergrond te laten werken. Zo zie je bijvoorbeeld het demo-programma in een venster terwijl je zit te kaarten. Vanzelfsprekend is het hele systeem enorm traag.
De mogelijkheden voor een speciale Atom-for-Windows worden inmiddels bestudeerd.

Onder OS/2 2.1 draait een en ander een stuk beter. In een DOS-box draait de Atom prima mee. Vanzelfsprekend is het systeem dan niet traag. Ik heb zelfs als experiment twee Atom kaarten in mijn PC geplaatst en deze beide in een DOS-box laten draaien. Dat ziet schitterend uit. Een icoontje voor de Atom in OS/2 is ook bijgevoegd op de diskette.

Bij zowel Windows 3.1 als OS/2 2.1 kunnen er foutmeldingen van het operating systeem verschijnen als de Atom naar een extended videomode wil omschakelen. Dit is te voorkomen door de Atom niet in een venster maar in een geheel scherm te laten starten.

Tenslotte is de Atom al getest in een omgeving met NetWare Lite van Novell. Via het netwerk kan de Atom probleemloos de harddisk, diskdrives en CD-ROM speler van een andere computer benaderen. (Misschien wordt het dan toch tijd om het Atom disk archief eens op CD te zetten)...

Dan had ik tenslotte nog een klein probleempje gezien als de Atom verlaten wordt met de opdracht *COMMAND. Tijdens het intikken van opdrachten (dus in MS-DOS) verschenen allerlei rare tekens willekeurig op scherm. Dit heb ik verholpen door de dos-driver ANSI.SYS niet in het upper memory te laden, maar gewoon laag. Daarna heb ik het probleem niet meer gezien.

Als er nog vragen zijn, bel of schrijf gerust,

Met de vriendelijke groeten van Roland.

**Er zijn helaas enkele gevallen bekend waarbij de software niet werkt met een 65C02 processor. Met name een R65C02 kloon en een G65SC02p-2 van CMD lopen vast in de Branquart schakelsoft. Het plaatsen van een andere processor loste in beide gevallen de problemen op.**

************************************************************************

# Eenvoudig 32Kbyte geheugen, van #0000 tot en met #7FFF, met backup en write protect.

************************************************************************

In A.N. jrg.7 nr.3 blz.20 stond een artikel over mijn plannen om een Atom te herbouwen.
Mijn hoop op reactie hierop is tot op heden niet beloond.
Mede omdat ik er dus alleen voor sta, komt het project maar langzaam van de grond.

Een van de eerste ( kleine ) proeven is echter enige tijd geleden met succes afgerond, en omdat deze ook als zelfstandige modificatie bruikbaar is, wil ik er toch melding van maken.

Het gaat hier, zoals de titel vermeldt, om het lage geheugen.
Dit zijn de geheugen IC's met de nummers 10 t/m 19, 51 en 52 op het mainboard, 16 t/m 21 op de originele FDC kaart, de complete 16K kaart en verder alles in het adresgebied van #0xxx tot #8xxx, zoals een voormalig blok #Exxx op de schakelkaart, dat naar #1xxx is omgebouwd.
Daarnaast spelen ook de address decoder IC's, met name de IC's 5, 6, 7 en 8 op het mainboard en IC 12 op de FDC kaart, een rol.
Al deze stroomvreters kunnen verwijderd en vervangen worden door slechts drie IC's en een tiental losse onderdelen.
Het #1xxx blok op de schakelkaart bouwen we terug naar #Exxx of geven het een functie op #Axxx ( misschien daarover een andere keer meer ).

Nu zult u denken:"Zo iets is al eerder vertoond", zie de BRAIN TRANSPLANT van W.Kautz en Th.Waayer in A.N. jrg.7 nr.3 blz.20.
Maar bij mijn weten is er nog geen beschrijving of schema met backup en write protect ( in blokken van 4K ) in A.N. verschenen.

Op een IC na kunnen we alles bouwen op een simpel printje dat aangesloten kan worden via de leeggekomen voeten of zoals ik gedaan heb, middels de in mijn Atom lege voet voor IC 24.
De door mij gekozen oplossing heeft het voordeel dat de meeste aansluitingen op de juiste plaats voorhanden zijn, zodat het printje met zo'n speciaal voetje met aan beide zijden pennetjes, dat we recht onder de 32K voet solderen, kan worden aan gesloten.
Van het 32K IC dienen slechts de pinnen 1, 2, 27 en 28 boven en buiten voet 24 om verbonden te worden met de maiboard en er hoeven maar enkele aansluitingen van voet 24 niet naar het 32K IC te gaan (zie opmerkingen aan het einde van dit artikel).

Het belangrijkste IC is de in brain transplant reeds gebruikte 43256 of 62256.
Een aardige weet, de laatste cijfers geven het aantal kilobit aan ( 32Kbyte * 8bit = 256Kbit ), dit geldt voor veel RAM IC's, bijvoorbeeld 2Kbyte 6116 en 8Kbyte 6164.

Voor het write protect gebeuren gebruik ik niet de voor de hand liggende 74LS138, maar een 74LS251, dit heeft namelijk het voordeel dat ik meerdere 4K blokken met een IC onder write protect kan schakelen.

Het derde IC is een 74LS10 dat op het mainboard de IC's 5, 6, 7 en 8 vervangt.

Verder is er de bekende backup schakeling, die een kleine aanpassing heeft ondergaan.

Dan nu het schema ( fig.1 ) en een korte beschrijving van de 32K met de write protect schakeling.

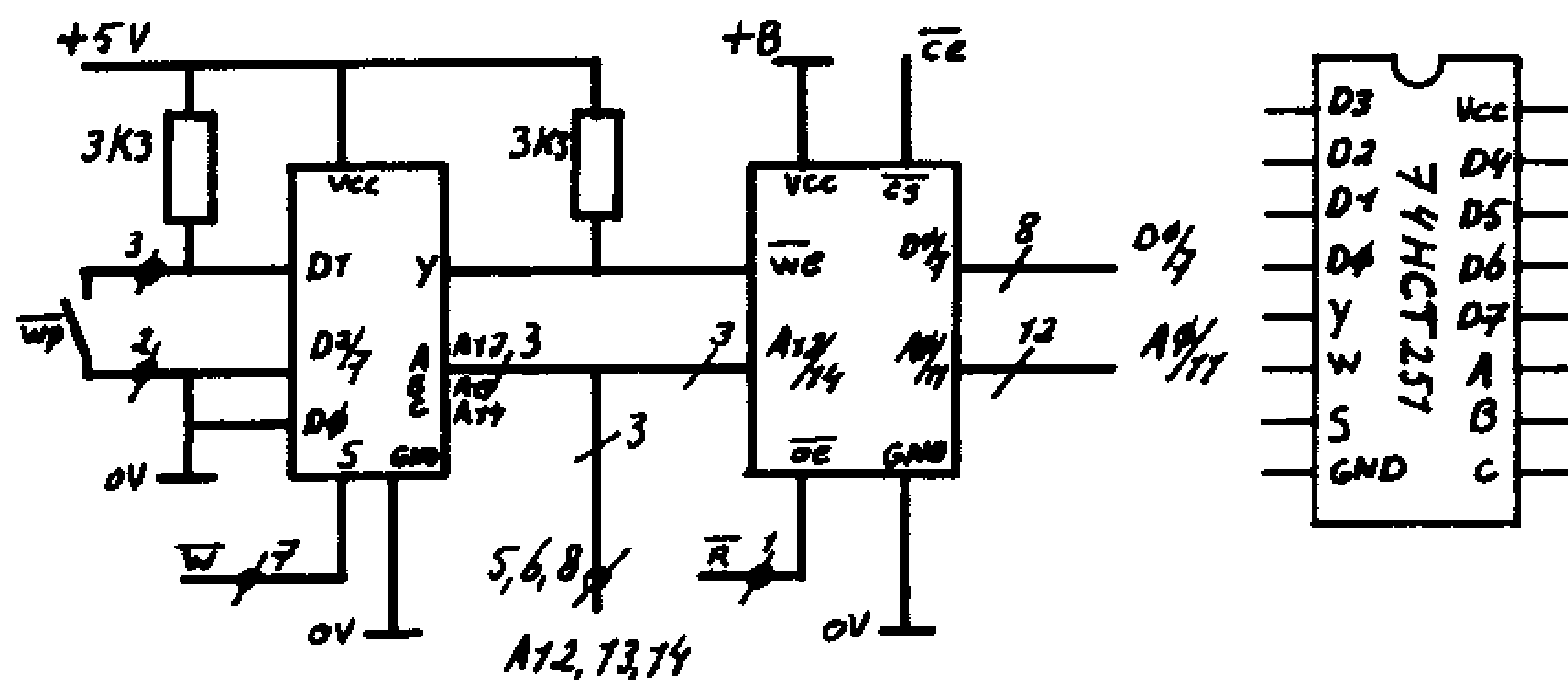



fig.1

De 74LS251 vangt met zijn S ingang het NWDS signaal op.
Door de adres lijnen 12, 13 en 14 wordt bepaald binnen welk 4K blok geadreseerd is, inderdaad wordt er bijvoorbeeld niet alleen op #1xxx gereageerd, maar ook op #9xxx, maar dat geeft niet want dan blijft ( A15 ) hoog en dus ook de NCE ingang van het geheugen dat dan niet kan reageren.
De ingangen D0/7 van de 74LS251 komen nu overeen met de geheugen blokken #0xxx/#7xxx.

Elk blok dat pertinent niet onder write protect moet, bijvoorbeeld de zero page ( #0000/#0FFF ), krijgt op de bijbehorende ingang ( in dit geval D0 ) een draadje naar aarde.
Van blokken die wel onder write protect moeten kunnen, geven we de Dx ingang een verbinding naar een schakelaar met pullup weerstand.
Hierbij geld dat als de schakelaar open is het blok protected is.

We kunnen dit op meerdere manieren uitvoeren, eenvoudig alle ingangen aan een schakelaar met weerstand ( gezamenlijke write protect ), of sommige ingangen een eigen schakelaar en weerstand ( dit voorkomt dat bij verkeerd adresseren bijvoorbeeld een machine code in plaats van op #7xxx op de Branquard belandt ).
Maar het is ook mogelijk om bijvoorbeeld blok #1xxx een een eenvoudig drukknopje met maakkontakt ( en een weerstand ) te geven als write enable schakelaar voor de schakelsoft o.i.d., blok #7xxx ook en de blokken #3xxx t/m #6xxx samen een aan/uit schakelaar ( en een weerstand ) te geven.
Er kan zelfs een schakelbyte op de ingangen D0/8 gezet worden !

De weerstand tussen de Y uitgang van de 74LS251 en de +5V is nodig omdat deze uitgang tri-state is ( hoog ohmig als ingang S hoog is ).

De aanpassing van de backup schakeling is vrij simpel ( fig.2 ).

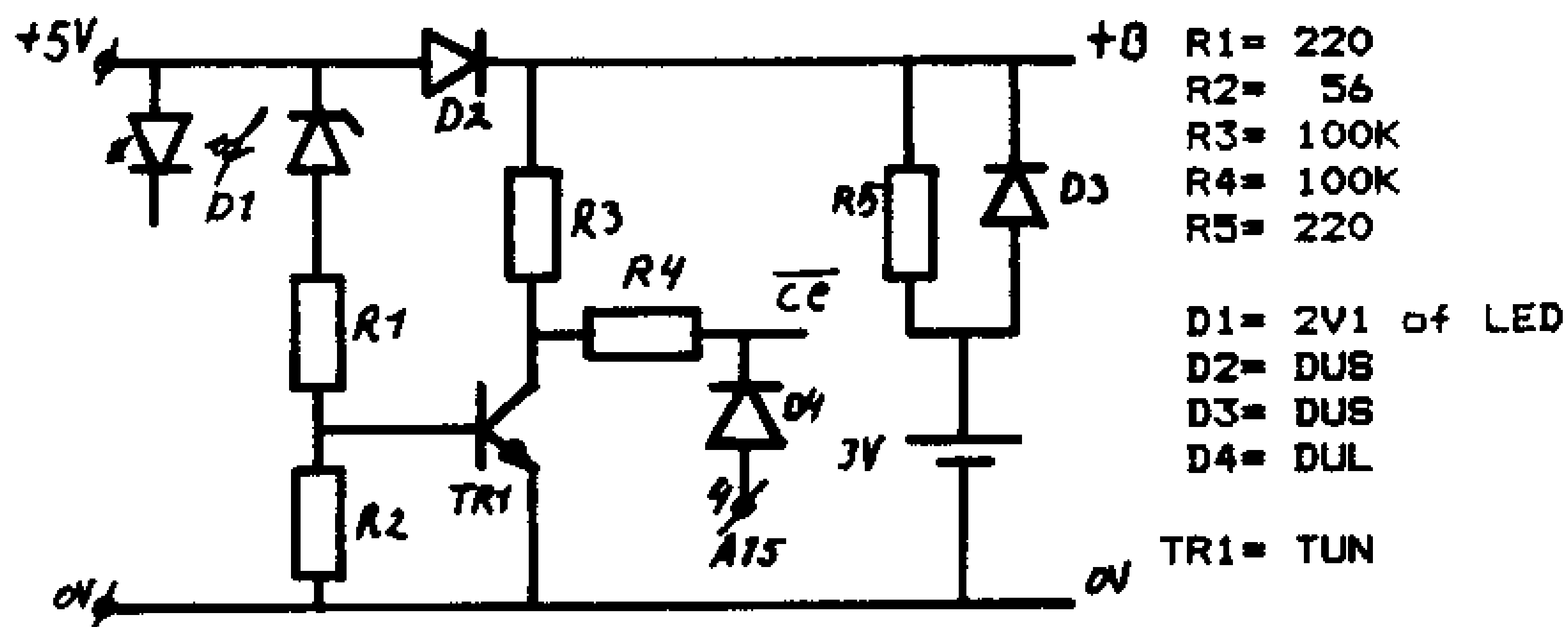


fig.2

De extra weerstand aan de collector van de transistor vormt samen met de schakeldiode een OF poort.
Als de adreslijn A15 of de backup hoog is, dan is het 32K IC disabled, eenvoudiger kan volgens mij niet.

De 74LS10 kan, als deze hier op voorbereid wordt, in de voet van het oude IC8 geplaatst worden.

Dit gaat als volgt:

Buig de pinnen 3, 4, 5, 8 en 13 op zijn buik.
Buig de pinnen 9, 10, 11 en 12 naar buiten.
Verbind pin 2 met pin 13 ( draadje over de buik ).
Verbind de pinnen 3, 4, 5 en 8 met elkaar ( draadje ).
Plaats het IC in voet 8, het heet nu ICn8.
Verbind ICn8 pin 9 met voet 49 pin 7 ( #BCxx, FDC ).
Verbind ICn8 pin 10 met voet 23 pin 13 ( #Axxx, boxen ).
Verbind ICn8 pin 11 met voet 8 pin 4 ( #Exxx, DOS ).
Verbind ICn8 pin 12 met voet 8 pin 3 ( #8/#9xxx, VDG ).
Draadbrugje tussen voet 5 pin 8 en voet 5 pin 11 ( I/O ).

Voor deze verbindingen gebruik ik wire wrap draad, dat is eenvoudig op zijn plaats te houden door het door enkele doorvoergaatjes in de print te vlechten, zo beland ik meteen aan de soldeerzijde van de voeten.

Op het door mij ontworpen printje gebruik ik een achtvoudig connectortje voor de extra verbindingen die niet op voet 24 voorkomen.

Dat zijn de volgenden:

Aansluiting 1 aan voet 25 pin 5 ( NRDS ).
Aansluiting 2 aan aansluiting drukknopje.
Aansluiting 3 aan andere aansluiting drukknopje.
Aansluiting 4 aan voet 22 pin 25 ( A15 ).
Aansluiting 5 aan voet 22 pin 22 ( A12 ).
Aansluiting 6 aan voet 22 pin 23 ( A13 ).
Aansluiting 7 aan voet 25 pin 36 ( NWDS ).
Aansluiting 8 aan voet 22 pin 24 ( A14 ).

Aanduidingen tussen haakjes corresponderen met fig1. en fig.2.

Nog enkele laatste opmerkingen:

Als u, zoals ik, voet 24 gebruikt, dient u ervoor te zorgen dat de aansluitingen 20, 21 en 24 van de voet helemaal niet met het geheugen IC verbonden worden en aansluiting 18 van de voet niet rechtdoor maar met pin 23 van het geheugen IC.

Let er op dat de op zijn buik gebogen pootjes van ICn8 geen kontakt maken met de voet waar het IC in geplaatst is.

Met "ic.. pin.." bedoel ik een aansluiting op een IC zelf, met "voet.. pin.."bedoel ik een aansluiting op een IC voet aan de soldeerzijde van de print.

Vergeet niet de aan het begin van dit artikel genoemde IC's te verwijderen.

Let er op dat alleen de 32K uit de backup gevoed wordt.

Het stroomverbruik onder backup schat ik op enkele micro ampere ( data retention current is typical 1 en maximum 50 micro ampere ).

Mocht u vragen, raad of kritiek hebben schrijf of bel dan naar:

```
>RUN
BEGIN ADRES?#FF60
INDIRECT ?  JA OF NEE?NEE

FF60   8A           TXA
FF61   8D  3 B0     STA  #B003
FF64   A9  7        LDA @# 7
FF66   8D  2 B0     STA  #B002
FF69   20 D1 F7     JSR  #F7D1
FF6C    6  C        ASL  # C
FF6E    F           ???  # F
FF6F   41 43        EOR (#43,X)
FF71   4F           ???  #4F
FF72   52           ???  #52
FF73   4E 20 41     LSR  #4120
FF76   54           ???  #54
FF77   4F           ???  #4F
FF78   4D  A  A     EOR  #A0A
FF7B    D A9 02     ORA  #02A9
FF7E   85 12        STA  #12
FF80   58           CLI
FF81   A9 55        LDA @#55
FF83   8D  1 29     STA  #2901
FF86   CD  1 29     CMP  #2901
FF89   D0  C        BNE  #FE97>
```

```
> RUN
atom DISASSEMBLER

HEX START ADDRESS?#FF60
END ADDRESS?#FF84
BRK/RTS OPTION?N
CODE STORAGE TEXT SPACE:
(EG. #29)?N
 FF60   8A         TXA
 FF61   8D 03 B0   STA #B003
 FF64   A9 07      LDA @#07
 FF66   8D 02 B0   STA #B002
 FF69   20 D1 F7   JSR #F7D1
 FF6C   06 0C      ASL #0C
 FF6E   0F         ???
 FF6F   41 43      EOR (#43,X)
 FF71   4F         ???
 FF72   52         ???
 FF73   4E 20 41   LSR #4120
 FF76   54         ???
 FF77   4F         ???
 FF78   4D 0A 0A   EOR #0A0A
 FF7B   0D A9 02   ORA #02A9
 FF7E   85 12      STA #12
 FF80   58         CLI
 FF81   A9 55      LDA @#55
 FF83   8D 01 29   STA #2901
```

```
>
>LOAD"VISLOAD"
PLAY TAPE
>L.
 10 REM VISLOAD 10-12-81
 12 REM ARIE MARCHAL
 15 DIM SS3;?#80=0
 16 FOR I=1 TO 2
 20 P=[illegible]
 30[PHP;PHA;STX#EC;STY#C3
 32 CLC;ADC@32
 35 STA#81;CMP@127;BCSSS0
 40 EOR@96;STA#81
 42 CMP@63;BCCSS1;CMP@#7F;BCSSS0
 44 LDA@32;STA#81
 45:SS0 CMP@#C0;BCC SS1
 46 LDA@32;STA#81
 50:SS1 LDX#80;BNE SS2
 52 LDA@32;LDY@0;:SS3 STA#8000,Y;INY;BNESS3
 56:SS2 LDA#81;STA#8000,X
 60 INC#80;PLA;LDX#EC;LDY#C3
 70 PLP;JMP#F[illegible];]
 75 N.I
 80 ?#214=#A0;?#215=#28
 84 P.'';?224=10
 85 P."visable load"''"DIT PROGRAMMA KAN NA RUN"''"OVERSCHREVEN"
 87 P." WORDEN."''"ZET NA EEN BREAK DE"''"VECTOR WEER GOED MET"'
 88 P.'"?#214=#A0;?#215=#28"''
 90 END
```

**Dat wilden we graag even kwijt!** *Intel Outside*

Wat nou zult u denken. Gaan we oubollig doen, als een fatsige man, wiens buik niet alleen boven de broekriem zit, maar er ook al overheen hangt, en wiens inhammen in zijn grijs begroeide bovenhoofd groot genoeg zijn om een redelijke middenklasser te kunnen keren? Nee. Dit is een stukje over een gigantisch project. Door slechts enkelingen bedacht, gemaakt en gerealiseerd. Kortom: voldoende om eens over uit te wijden

**De kaart**

Begin januari was het zover: de atomkaart is in het Harderwijkse gearriveerd. Een kaart van grote klasse. De print is dubbelzijdig uitgevoerd, keurig doorgemetaliseerd en alles is keurig vertint. De connector, die zijn wortels in de PC gaat krijgen, is verguld. Hulde! Zo hoort dat, om ook nog volgend jaar en het jaar daarna, plezier van deze kaart te hebben.
De bijgeleverde handleiding laat aan duidelijkheid niets te wensen over. Alles staat er helder in en, voor zover het niet duidelijk is, wordt men uitgenodigd de vader van deze kaart, persoonlijk te consulten.
Een lijstje waar de onderdelen op beschreven staan ontbreekt, maar het schema is dermate open, dat het eenvoudig valt af te leiden. De setting van de jumpers is even zoeken, maar dat kan komen doordat mijn A4tje-layoutopstelling een kopie is, gemaakt op een kopieermachine met een matige signaal/ruis verhouding.

**De bouw**

Tenzij u slechts als loodgieter ervaring heeft met het solderen van dakgoten, is het voor iedereen die de Atom nog steeds aan de praat heeft mogelijk deze kaart te bouwen. Je moet van goede huize komen om iets fout te doen. Bepaald een geruststellende gedachte voor hen, die hier toch wel wat tegenaan hikken.
Daar er nog een restje van de dertiende maand niet naar de belasting bleek te gaan, heb ik me laten verleiden tot de aanschaf van een 65802, een chip als de 6502, maar met wat extra instructies. Best wel een rib uit je lijf, maar ik ben al jaren verliefd op dat ding, dus nu had ik de kans om daar handen en (ie)voeten aan te geven.
Bij ondergesprekende verliep, qua bouw, alles voorspoedig. Behalve met dat ene chippie: de

74HCT393. Dat is een deler. Nou in mijn geval begon dat ding wel heel letterlijk te delen. Wat was het geval: met mijn tulpenhooft plaatste ik dat ding verkeert om. Nee, niet met de pootjes omhoog, maar met het nokje naar beneden. Gek he, dat je dan geen clock signaal overhoudt. Maar goed. Nieuwe 393 gehaald, althans: "Op", dat zei die man achter de balie. Even voor u bestellen. Komt u over een week maar terug.
Nadat de 393 nu op een correcte wijze was geplaatst, ging het met de clock een stuk beter. De kaart deed het echter nog even niet. Na wat speurwerk en contact met de maker, bleek er een Gal niet te functioneren. Niet verwonderlijk, met al dat water in Limburg, dat ook een gal vol was gelopen. Maar met de snelheid van het licht, is er een nieuwe gal uitgeleverd en de kaart werkt. En hoe...

**De werking**

Nou, dat de kaart werkt, dat is zeker. De kwaliteit die zichtbaar is op de kaart, is ook in de software zichtbaar. Het is bepaald indrukwekkend als je ziet dat voor een eerste produktie serie (ik hoop trouwens dat er nog veel series zullen volgen), de hardware en de software zonder problemen werkt. Daar kunnen de grote bedrijven, (sorry Bill, Ik Ben Maf) nog wat van leren.
Het is verbazingwekkend, dat zowel de hardware als de systeemsoftware door dezelfde man zijn gemaakt. Dat zie je niet vaak, dat die twee kwaliteiten in 1 mens zitten.
De software werkt heel prettig. Alles is nu zeer eenvoudig menu gestuurd te maken. Het laden van een bestand is een feest. Tenopzichte van die floppen is de harde schijf toch echt een feest. Wat alleen opvalt: mijn pakket Microsoft Excel (Hi Bill) is goed voor zo'n 8 Megabyte. Als ik al m'n Atom software neem, moet ik persen, overtrekken, overdrijven, kortom de waarheid geweld aandoen, om te zeggen dat ik 2 Megabyte aan Atomsoftware heb. Het neemt een nietig plekje in op de harde schijf. Laat staan als ik de zaak met PKZIP in elkaar pers. Dan zou het wel eens allemaal op 1 HD diskette passen....
Nooit gedacht trouwens, dat ik vanuit m'n Atom ooit vals fluitend een CD-Rom kon benaderen. Nu nog op zoek naar een CD-Rom vol met Atom software.

**Conclusie**

De Atom in PC kaart is een goed werkend geheel. De federatie heeft een goede job gedaan: Zij hebben de nek uitgestoken voor een project, waar de impact nauwelijks van was in te

schatten. Toch, gelukkig, ze hebben het gedaan.

Maar het is nauwelijks voorstelbaar wat deze kaart werkelijk zou moeten kosten. Als ik een ruwe schatting van de inspanning maak, denk ik dat ik voorzichtig ben als ik zeg, dat in de soft- en de hardware zo'n 800 manuur zit. Als je dit door mensen laat vervaardigen, die normaal gesproken de boterham verdienen met dit soort activiteiten, dan zit je op zo'n 160 gulden per uur. Vermenigvuldig dat met die 800 uur. Verdeel dat eens over een serie van 25 printen. Tel uit je winst. Toch nog maar iets extra's overmaken op giro 777 voor Limburg.

Feitelijk onbegrijpelijk dat de print slechts 100 gulden kost en dat je de rest voor een paar kralen en een spiegel erbij krijgt. Om over de subsidie van 50% op de kaart nog maar te zwijgen.

En valt er nu niets meer te wensen? Nou, eigenlijk niet. De kaart is gewoon af, de software OK. In de toekomst kan de software nog verder worden uitgebouwd, maar dat geldt voor alles in dit leven.

Als ik nog een wens zou mogen doen, dan zou ik graag een interrupt door deze kaart willen laten genereren bij het schrijven naar Atom videogeheugen. Maar ik geef toe dat ik een maar een miezerige zeur ben, want voor de huidige software is dat volstrekt overbodig, maar voor de Atom in een multitasking omgeving (Windows), zou het me niet slecht uitkomen als die kaart me vertelde, dat ik de rest van het systeem even op de handrem kan zetten. Volkomen gezeur dus, spijkers op laag water (hmm... slecht voorbeeld) zoeken.

De federatie levert een fraai stuk werk af. Als je ziet hoe het probleem met de gal is opgelost, dan kan menige (lees: iedere) service organisatie daar een voorbeeld aannemen: Probleem wordt vastgesteld, oplossing gevonden, oorzaak gezocht en patches naar eindgebruikers gestuurd. Dit alles zonder uitvoerige correspondentie, of dreiging van opzeggen van contract, of zware gesprekken met tjonge-wat-stinkt-die-sigaar rokende directie. Nee, gewoon geen gezeur. We lossen het op. Binnen een week in huis.

En hoe nu verder...

Voorstel is om de gehele harde kern federatie op te dragen voor een lintje en Roland te benoemen tot Ridder in de Federatie, minimaal ere lid te maken.

Een ieder die deze kaart nog niet heeft, moet eens bij zichzelf te rade gaan of er nog niet ergens een oude sok, een suikeroom of tante ligt, of dat de bestemming van het vakantiegeld voor dit soort doelen kan worden aangewend. Deze kaart is namelijk niet zomaar een vervanger van de Atom, maar stijgt uit boven dat niveau. De potentie van de Atom in zijn geheel en die van deze kaart in het bijzonder, is groot. Het is aan ons allen dat verder uit te bouwen in hard- en software. Dame en heren: aan het werk, de Atom is er!

Leendert Bijnagte

Atom Inside

Onder deze kop ga ik u de komende tijd vermoeien. Wat ik in dit stuk wil doen, is het beschrijven van wat er zoal komt kijken, of kan komen kijken, bij het brengen van de PC-Atom kaart onder Windows.
Sta mij toe u eerst iets over Windows of eigenlijk PC's in zijn algemeenheid te vertellen.

Waarom Windows. Eigenlijk kort en simpel: omdat MS-Dos niet echt een operating systeem is (sorry Bill). MS-Dos is nooit opgezet om meerdere taken gelijktijdig te doen. Dat tekort is feitelijk min of meer aangevuld door MS-Windows. Dit Windows neemt in feite de gehele machine over en kan zo een aantal taken doen die MS-dos niet kan.
Nu zal de echte kenner zeggen dat ook Windows niet echt een multitasking operating-systeem is, en ze hebben gelijk. Als er meerdere programma's lijken te draaien dan begint het feitelijk op een estafetteloop te lijken. Stel dat we een tekstverwerker, een spreadsheet en een spelletje hebben draaien. Wat er nu gebeurt, is het volgende: taak 1 (tekstverwerker) krijgt het estafetestokje en als het hem gelegen komt, geeft deze het stokje weer aan het besturingssysteem (Windows), vervolgens geeft deze hem door aan taak 2 (spreadsheet). Idem dito voor taak 3 het spel. Het laat zich raden wat er gebeurt als taak 2 het stokje niet teruggeeft: Dan houdt het feest op! De rest staat dus feitelijk in bevroren toestand!

Maar dan de PC-Atom. Dat is even iets anders. Dat is een kaart waar een volledig eigen proces op loopt. Wat is er nu leuker om dit met Windows te combineren? Gewoon even een programmaatje maken die alles wat de Atom doet, in een window laat zien.
Toch gaat dat niet zomaar vanzelf, daar is toch het nodige voor nodig, inmiddels door ervaring wijs geworden had u natuurlijk niets anders verwacht. Om u vast op te warmen, wat overdenkingen: Windows heeft een eigen besturingschil, een zogeheten kernel, die het niet zondermeer toestaat dat je I/O gaat doen. Doe je toch, dan wordt je door Windows in de rug geschoten en gaat je programma naar de eeuwige jachtvelden. Om toch I/O te kunnen doen, moet er een Windows device driver worden ontwikkeld. Hier heb je de Windows Device Driver Devellopment Kit nodig. Gelukkig heb ik die, dat wil zeggen mijn werkgever, maar dat is bijna hetzelfde.

Windows kent heel veel lettertypes. Maar de Atom als lettertype bestaat niet. Geen probleem, dan maken we er zelf een. Dus eentje met alles er op en er aan. Dat is inmiddels redelijk afgerond. Het ziet er als volgt uit: De normale uppercase ASCII, op de plaats van de kleine letters de geïnverteerde hoofdletters, de blokjes die we kennen uit clear 0, en, omdat er toch nog wat ruimte over was, heuse kleine letters. Gewoon Hex 80 bij een kleine letter optellen en het is een echte kleine letter. Inmiddels zijn er twee uitvoeringen van: de Bitmap en een Truetype. Truetype wil zeggen dat het teken niet als bitmap wordt opgeslagen maar als beschrijving. Neem als voorbeeld de letter O. Deze wordt ongeveer opgeslagen als de omschrijving: Twee ellipsen, waarbij de buitenste iets groter is dan de binnenste. De ruimte ertussen is opgevuld. Het voordeel is dat het op ieder beeldscherm of printer een fantastische kwaliteit geeft. Dat het een bereklus is om zoiets te maken, dat laat zich raden. Maar ja, hoeveel tijd zit er inmidels in onze Atom?

Wat is er verder allemaal mogelijk? Het is de bedoeling dat het geheel volledig compatibel is met wat de Atom kan. In principe kunt u dus Invaders in een window draaien. Toetsen worden netjes afgevangen. Hoe het staat met het pollen van #b001 op een shifttoets, dat wordt nog even testen, maar omdat de hele handel toch in ram draait moet ook daar wel iets op mogelijk zijn. Het windowtje waarin dit alles geschiedt, is feitelijk een monitor op de 6847 chip. Hoe het qua snelheid gaat weet ik nog even niet.

Verder kunt u alles doen wat Windows u toestaat. Het maken van een screendump kan dus gewoon met een optie uit het menu, Windows regelt dat dan verder. Eigen screendumproutines zijn dus niet meer nodig. Net als knippen en plakken. Via het clipboard kunt u gegevens uit de Atom halen. Geinig he?

Maar er is nog veel meer mogelijk, het moet allemaal nogwel gemaakt worden maar denk maar eens met me mee:

Listing overhevelen van de Atom naar bijvoorbeeld een krachtige editor, en vervolgens weer terugsturen naar de Atom. Of meerdere windows, een voor de textmode en een voor de grafische mode. Gewoon even softwarematig #8000 en #8100 verhuizen. Of wat denkt u van de Floating point rom vervangen door een stukje software wat met de floatingpointprocessor van de PC babbelt. Kan zo tot 1000 x sneller verlopen, bijvoorbeeld op een 486 33Mhz machine.

U ziet: de PC is the limit, niet de PC-Atom.

Maar wat moet er nou allemaal nog gebeuren, voordat het zover is?

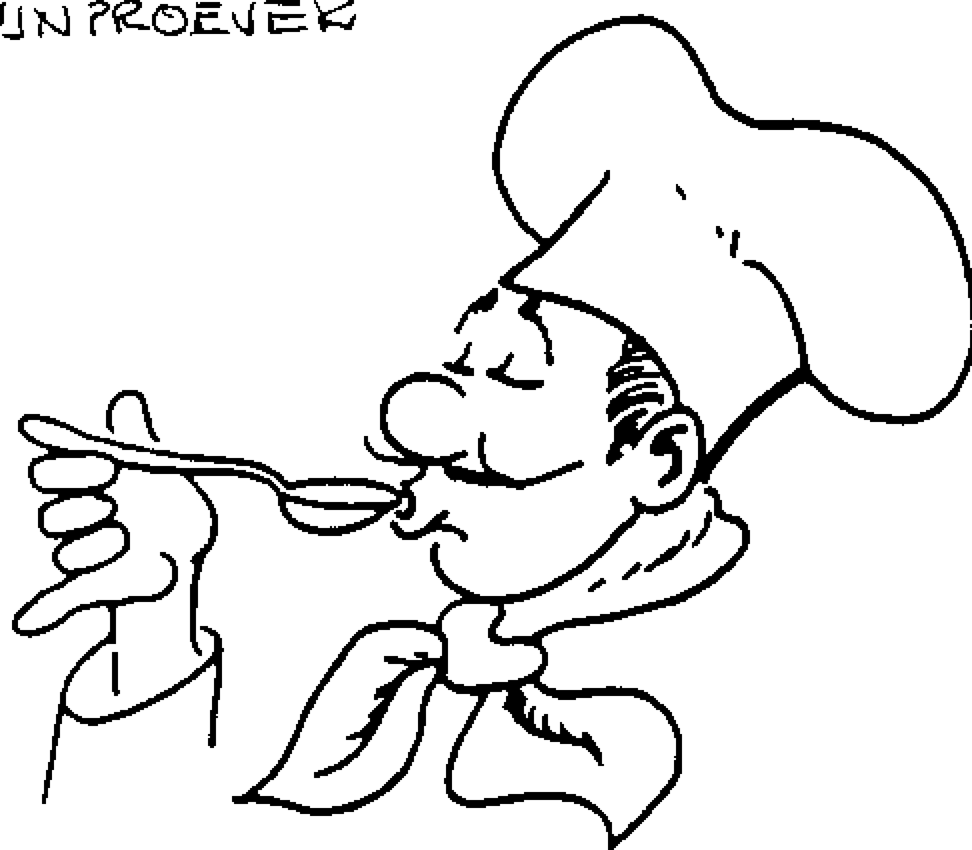
Best wel veel. Met name in de vrije uurtjes. Het ontwikkelen van een devicedriver is iets

wat voornamelijk in assembler moet. Nou is dat nog niet zo erg, maar met name onder een besturingssysteem als Windows, is het fouten zoeken erg lastig want om de klipklap hangt Windows. Niet omdat het zo slecht is, nee, maar omdat devicedrivers zo diep ingrijpen op een besturingssysteem.

Het hele client stuk van de Atom, dus het window wat alle Atomzaken laat zien, leeft op deze driver. Als de driver eenmaal af is, dan is het ergste leed geleden. Er moet nog iets slims worden verzonnen voor het snel kopiëren van de inhoud van het videogeheugen van de atom naar een Windowsbitmap, maar daar moet uit te komen zijn. Dit soort programmatuur wordt voornamelijk in C geschreven, maar laat zich gelukkig makkelijk testen. Toch geldt ook hier dat Keulen en Aken niet op een dag zijn gebouwd, of zo u wil Atom en PC. Het zal dus een produkt worden wat van grof naar fijn zal worden ontwikkeld! Uiteindelijk uitkomen op een programma van 6000 regels, is denk ik wel wat er zal gebeuren.

Ik hou u op de hoogte en wie interesse heeft om met me mee te denken of om te helpen of om te helpen testen: laat het me weten!

Leendert Bijnagte



# 6502 VERSUS 65C02

## door roland leurs

Sinds de Atom-in-PC kaart ben ik aan de slag met een R65C02 microprocessor voor de Atom. Door wat probleempjes bij het programmeren van de 65C02 bleken er opeens heel wat meer verschillen tussen beide processoren te zitten dan ik aanvankelijk dacht. In dit artikel staan eens een aantal punten opgenoemd waarin die verschillen aangegeven worden.

De 65C02 is een C-MOS uitvoering van de 6502; er zijn wat instrukties toegevoegd en de 65C02 kent nog een extra adresseringsmode, de indirecte adressering. Voorbeelden van nieuwe instrukties zijn bijvoorbeeld PHX en PHY (om het x resp. y register op de stack te zetten) en een voorbeeld van de indirecte adressering is LDA (#80).

Maar er zijn ook fouten uit de 6502 verbeterd in de 65C02.

- Alle onbekende opcodes uit de 6502 zijn nu gewijzigd in NOP instrukties; alleen het aantal cycli van de instruktie is niet altijd hetzelfde. Verder is het niet gedefiniëerd hoeveel bytes een dergelijke NOP lang is. Dat houdt dus in dat de program pointer een onbekend aantal plaatsen vooruit gaat.

- De indirecte sprong is gewijzigd; als het sprongadres net op een pagina grens ligt wordt het hoge byte van het adres wel goed gelezen.
  Zie voor een toelichting en toepassing het artikel 'De microprocessor identificeren' elders in dit nummer.

- In decimale rekenmode werken de vlaggen N(egative), O(verflow) en Z(ero) nu wel correct. De D(ecimal)-vlag is na een reset gewist. Voor de 6502 is deze vlag ongedefiniëerd na een reset. De 65C02 staat na een reset altijd in de binaire rekenmode.

- De D(ecimal)-vlag wordt ook na een interrupt altijd gereset zodat een interruptroutine altijd in binaire mode gestart wordt. Na de RTI instruktie wordt het statusregister van stack gehaald zodat de D-vlag voor het onderbroken programma weer goed staat.

Ten gevolge van deze wijzigingen zijn de betrokken instrukties soms een klokcyclus langer. Tijd-kritische routines kunnen dus verschillende resultaten hebben op een 6502 en een 65C02.

Niet alleen is er gesleuteld aan de instrukties, maar ook aan de poten is gewerkt. Zo zijn er versies die pin 5 gebruiken als /MemoryLock signaal. Tijdens een read-modify-write instruktie, daarbij wordt data gelezen-aangepast-geschreven, wordt deze lijn actief zodat andere processoren in het systeem niet op de bus kunnen.

De 65C02 antwoordt op de RDY-ingang zowel tijdens lezen als schrijven. Deze ingang dient om de processor even te stoppen als er langzaam geheugen wordt aangesproken.
Ook op een reset reageert de 65C02 anders dan de 6502. De 65C02 zet namelijk de program counter en het statusregister op de stack.

In zijn geheel heeft de 65C02 een veel lager stroomverbruik en veel minder warmteproductie ten opzichte van de 6502.

Deze en vele andere zaken staan in het boek

*Programming the 65816, including the 6502, 65C02 and 65802*
*door David Eyes en Ron Lichty*
*Uitgegeven door Prentice Hall Press*
*ISBN 0-89303-789-3*

Een nadeel van de 65C02 is dat er te veel fabrikanten zijn die deze processor maken. Er blijken in de praktijk veel verschillen te zijn tussen 65C02 processoren onderling. Dat maakt hem niet universeel. Er zijn zelfs C-MOS processoren die de mis in gaan op standaard 6502 instrukties!

Met de vriendelijke groeten van Roland.

# DE MICROPROCESSOR IDENTIFICEREN

## door roland leurs

Iedereen weet dat de 65C02 een verbeterde en uitgebreidere versie is van de 6502. In de 6502 zitten enkele fouten die in de 65C02 gewijzigd zijn.

Een van die fouten is de indirecte sprong waarbij het adres net op een paginagrens ligt, dus

JMP (#32FF)

De 6502 maakt hierbij een rekenfoutje en leest het lage byte van het sprongadres uit #32FF en het hoge byte uit #3200 (terwijl dat uit #3300 gelezen zou moeten worden). In de 65C02 is dat foutje weggewerkt.

Met behulp van dit verschil kunnen we dus eenvoudig bepalen met welke processor we in onze Atom te maken hebben:

```
10 REM TEST PROCESSOR
20 P=#3280;[
30 JMP (#32FF)
40 PHA;JSR #F7D1;]
50 $P="Deze Atom heeft een 65";P=P+LEN(P);[ NOP
60 PLA;JSR #FFF4;JSR #F7D1;]
70 $P="02 microprocessor.";P=P+LEN(P);[
80 NOP;JSR #FFED;RTS;]
90 P=#3310;[;LDA @CH"C";JMP #3283;]
100 P=#3210;[;LDA @#06;JMP #3283;]
110 ?#32FF=#10;?#3300=#33;?#3200=#32
120 LINK #3240;END
```

Dit programma zorgt ervoor dat er altijd een gedefiniëerde sprong plaats vindt, of het hoge byte nu uit #3300 of uit #3200 gelezen wordt. Afhankelijk van de sprong, en dus de processor, heeft de accu een bepaalde waarde bij de terugkeer op adres #3283 (regel 40).

Lees voor meer verschillen ook eens het artikel "6502 versus 65C02", elders in dit nummer.

Met vriendelijke groeten,
Roland Leurs

IntelCat File/Ext Report for LC.DB - 5:42 pm Mar 16, 1994 PPg 1

| File Name | Disk Name | Size | Date | Time |
|---|---|---|---|---|
| 3D-4-OP. | AN.94-01 | 4,096 | 10/04/93 | 05:00:34 pm |
| 4OP1RY. | AN.94-01 | 2,821 | 10/04/93 | 05:19:58 pm |
| ASTFIRE. | AN.94-01 | 5,371 | 08/24/93 | 03:11:24 pm |
| ASTROLO. | AN.94-01 | 11,491 | 09/15/93 | 09:24:12 am |
| ATKEY. | AN.94-01 | 271 | 11/01/93 | 11:09:36 am |
| ATOMDEMO.HLF | AN.94-01 | 781 | 10/12/93 | 10:23:14 am |
| ATOMDEMO.RUN | AN.94-01 | 10,216 | 11/03/93 | 11:27:12 am |
| ATOMDEMO.S1 | AN.94-01 | 3,331 | 10/11/93 | 02:00:04 pm |
| ATOMDEMO.S2 | AN.94-01 | 6,391 | 10/11/93 | 02:24:28 pm |
| ATOMDEMO.S3 | AN.94-01 | 5,626 | 10/12/93 | 12:08:30 pm |
| ATOMICA. | AN.94-01 | 3,331 | 08/16/93 | 02:15:20 am |
| ATOMIDX. | AN.94-01 | 77,776 | 09/02/93 | 07:16:30 pm |
| BACKGAMM.RUN | AN.94-01 | 4,606 | 10/13/93 | 06:17:26 pm |
| BANDIT. | AN.94-01 | 2,311 | 09/10/93 | 11:07:10 am |
| BATSHIP. | AN.94-01 | 4,861 | 08/24/93 | 11:02:40 am |
| BB-BALL. | AN.94-01 | 271 | 09/05/93 | 04:15:40 am |
| BINGO75. | AN.94-01 | 1,036 | 09/15/93 | 11:00:16 am |
| BINGO90. | AN.94-01 | 1,036 | 09/15/93 | 11:07:56 am |
| BLACKJAC. | AN.94-01 | 4,861 | 08/12/93 | 11:10:38 pm |
| BREAKOUT. | AN.94-01 | 3,076 | 08/17/93 | 11:02:20 am |
| BREAKOUT.RUN | AN.94-01 | 3,076 | 08/12/93 | 09:00:02 pm |
| CITYBOM. | AN.94-01 | 6,136 | 09/10/93 | 01:18:18 am |
| CLOCK. | AN.94-01 | 1,546 | 07/27/93 | 05:21:52 pm |
| CYLON. | AN.94-01 | 5,116 | 09/04/93 | 06:11:10 pm |
| CYLONN. | AN.94-01 | 5,371 | 09/08/93 | 07:16:04 pm |
| DARNIT. | AN.94-01 | 5,881 | 10/05/93 | 09:19:56 am |
| DARNIT.RUN | AN.94-01 | 5,881 | 09/26/93 | 12:23:44 pm |
| DARNIT2. | AN.94-01 | 5,881 | 02/09/94 | 06:07:44 pm |
| DOBBEL. | AN.94-01 | 4,606 | 01/18/94 | 05:22:18 pm |
| DODGEM2. | AN.94-01 | 1,291 | 08/17/93 | 10:01:18 am |
| DOMINO. | AN.94-01 | 5,116 | 02/10/94 | 11:10:50 am |
| DRPOKER. | AN.94-01 | 5,881 | 09/12/93 | 01:04:44 pm |
| DRUPPEL. | AN.94-01 | 3,841 | 09/15/93 | 05:14:00 pm |
| FIGHTER. | AN.94-01 | 5,137 | 08/10/93 | 09:15:58 am |
| FRACTAL. | AN.94-01 | 526 | 09/15/93 | 05:16:24 pm |
| GAGSDEMO. | AN.94-01 | 2,566 | 06/17/93 | 07:27:44 pm |
| GALAXBB. | AN.94-01 | 4,881 | 09/19/93 | 10:18:48 am |
| HIGHWAY. | AN.94-01 | 4,861 | 10/14/93 | 07:02:08 pm |
| INVADER.RUN | AN.94-01 | 5,371 | 10/02/93 | 11:21:56 am |
| INVADERS. | AN.94-01 | 5,116 | 08/15/93 | 04:18:52 pm |
| JOYEMUL.RUN | AN.94-01 | 781 | 10/12/93 | 07:23:36 pm |
| KASJE. | AN.94-01 | 3,586 | 01/09/94 | 10:24:08 am |
| KIENEN. | AN.94-01 | 4,606 | 08/17/93 | 11:17:06 am |
| KLFRUIT. | AN.94-01 | 3,841 | 09/16/93 | 04:05:52 pm |
| KLOK. | AN.94-01 | 271 | 07/12/93 | 09:23:48 pm |
| KLOKSKE. | AN.94-01 | 22,456 | 08/02/93 | 10:08:10 pm |

IntelCat File/Ext Report for LC.DB - 5:42 pm Mar 16, 1994 PPg 2

| File Name | Disk Name | Size | Date | Time |
|---|---|---|---|---|
| MASTER. | AN.94-01 | 5,626 | 06/13/93 | 11:15:30 pm |
| MATCHES. | AN.94-01 | 781 | 09/15/93 | 04:27:46 pm |
| MUISDEMO. | AN.94-01 | 526 | 07/06/93 | 08:20:04 pm |
| PIRAMIDE. | AN.94-01 | 4,606 | 03/06/93 | 08:00:04 pm |
| PIRAMIDE.RUN | AN.94-01 | 4,606 | 09/25/93 | 07:18:50 pm |
| PLAKOTO. | AN.94-01 | 5,116 | 01/12/94 | 11:18:00 am |
| PLANE. | AN.94-01 | 781 | 09/14/93 | 01:08:38 pm |
| PLANETS. | AN.94-01 | 526 | 08/13/93 | 12:26:58 am |
| PLANETS.RUN | AN.94-01 | 526 | 07/03/93 | 03:09:56 pm |
| PLANTAR. | AN.94-01 | 21,436 | 08/10/93 | 12:19:04 pm |
| POKER. | AN.94-01 | 5,881 | 08/12/93 | 11:00:24 pm |
| PONTOON. | AN.94-01 | 6,136 | 08/03/93 | 09:17:10 pm |
| RAMTEST. | AN.94-01 | 1,801 | 06/22/93 | 07:12:38 pm |
| REVERSI. | AN.94-01 | 2,311 | 08/22/93 | 11:25:10 am |
| REVERSI.RUN | AN.94-01 | 2,311 | 10/15/93 | 07:15:32 pm |
| ROMTEST. | AN.94-01 | 1,779 | 06/22/93 | 06:20:48 pm |
| ROTFAN. | AN.94-01 | 1,036 | 09/11/93 | 05:01:52 pm |
| SATURN. | AN.94-01 | 4,096 | 09/06/93 | 03:23:58 pm |
| SCHUIF. | AN.94-01 | 781 | 09/16/93 | 04:23:06 pm |
| SHOOT. | AN.94-01 | 3,331 | 01/11/94 | 11:16:14 am |
| SHUTTLE. | AN.94-01 | 2,566 | 01/09/94 | 11:18:52 am |
| SNAPPER. | AN.94-01 | 5,116 | 07/05/93 | 11:01:40 pm |
| SNAPPER.RUN | AN.94-01 | 5,371 | 10/12/93 | 08:18:50 pm |
| SPIDER. | AN.94-01 | 526 | 08/22/93 | 09:14:06 am |
| SPINNEWE.RUN | AN.94-01 | 526 | 10/15/93 | 06:05:34 pm |
| TRAP. | AN.94-01 | 5,371 | 09/12/93 | 09:19:44 pm |
| VALINET. | AN.94-01 | 4,861 | 08/21/93 | 05:28:16 pm |
| VALTIE. | AN.94-01 | 4,861 | 08/21/93 | 05:30:16 pm |
| W-WSHOW. | AN.94-01 | 1,546 | 08/25/93 | 02:27:12 pm |
| WARNING. | AN.94-01 | 3,841 | 08/25/93 | 05:09:34 pm |
| WIGGEL. | AN.94-01 | 1,546 | 08/10/93 | 10:30:12 am |
| WIGGELJ. | AN.94-01 | 1,801 | 01/28/94 | 09:04:16 pm |
| WIGGLEJ. | AN.94-01 | 1,801 | 10/04/93 | 01:17:22 pm |
| YAHTZEE. | AN.94-01 | 4,606 | 08/06/93 | 11:29:18 pm |
| ZEESLAG. | AN.94-01 | 4,861 | 01/12/94 | 09:05:54 am |
| ZOEK.RUN | AN.94-01 | 1.291 | 10/15/93 | 05:27:58 pm |

==================================================================
*                                                                *
*                                                                *

# A.C.C.-LAND-DAG VOORJAAR 1994

*                                                                *
==================================================================

DATUM : 23 APRIL 1994 ,van ca. 10.00 tot ca.17.00 u.
PLAATS : PAROCHIECENTRUM ,Melkweg 5,DE BILT.

Op de A27 de afslag Utrecht-Oost,maar let vooral op de kleine witte aanduiding "VEEMARKT", de Biltse Rading afrijden tot de splitsing Groenekan/Bilthoven ,hier r.a. ,20 meter verder l.a.
Voor busreizigers : Vanaf Utrecht CS lijn 57,stopt op de Melkweg
Breng brood mee, voor koffie wordt gezorgd.
Voorwaarde : Iedereen is zelf verantwoordelijk voor zijn eigen apparatuur ,etc.

DEMO's: o.a.
* R.Leurs : ATOM IN PC, verbeterde versie
* H.Bastings : Nieuwe videokaart met GAL
* ACCL. PL8/VIA: eenvoudige I/O-toepassing
* J.Geene : De kleurenkaart zonder ruis
* D.Bronsdijk ?

Het Bestuur van de Federatie.

==================================================================

AAN DE BESTUREN VAN DE ATOM COMPUTERCLUBS NEDERLAND BELGIE.

Uitnodiging voor de Algemene Veradering van de F.A.C.

Plaats : Parochiehuis Melkweg 5 , de bilt.
Tijd : Zaterdag 23 april 1994, te 11.30 u.

Agenda :
1. Opening door de voorzitter.
2. Notulen van de vorige vergadering.
3. Verslag van de penningmeester over 1993.
4. Verslag van de kascommissie, benoeming nieuwe kascommissie voor 1994.
5. Budget 1994.
6. Bestuursverkiezing : De volgens rooster aftredende bestuursleden zijn herkiesbaar.
7. Stand van zaken en herindeling van de regio's op grond valedental en wensen.
8. Vaststelling activiteiten 1994.
9. Wat verder ter tafel komt.
10. Rondvraag en sluiting.

De secretaris.

REGIO-ADRESSEN.

Wilt U lid worden van de ATOM COMPUTER CLUB ?.
Neem dan contact op met de penningmeester van de regio waar U bij ingedeeld wenst te worden. Deze kan U inlichten omtrent het lidmaatschap.

Regio NOORD-HOLLAND :
P.v.Kuik, Zuideinde 54-a, 1843 JP Groot Schermer.
tel. 02997- 1902.

Regio DEN HAAG +ARNHEM
Th.Waayer, H.v.Boeijenlaan 66, 2273 DC Voorburg.
tel. 070-3862504.

Regio BRABANT-OOST + ZEELAND
J.Teulings, K.Doormansstraat 54, 5224 GL Den Bosch.
tel. 073-212888.

Regio LIMBURG + BELGIE EN OOST/NOORD NEDERLAND
C.Rutkowski, Overhoven 80, 6136 EG SITTARD.
tel. 046-522673

Leden die behoren tot opgeheven regio's, danwel regio's die conform de statuten geen lid meer zijn van de federatie, worden in verband met de financ. administratie en de verzending van ATOM-NIEUWS,door de federatie toegewezen aan de nabije regio's.
Zo men tegen deze indeling bezwaar heeft, om welke reden dan ook, kan men de regio van eigen keuze opgeven aan de penningmeester van de federatie: T.Rutten , zie de pagina van de federatie in dit blad.

Bij het aangaan van het lidmaatschap kunt U de contributie overmaken op de rekening van de federatie. Vermeld hierbij uw volledige naam, en adres , alsmede evt. de regio waarbij U wenst te worden ingedeeld.